

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2001-109742
(P2001-109742A)

(43) 公開日 平成13年4月20日 (2001.4.20)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード*(参考)
G 0 6 F 17/21	5 0 1	G 0 6 F 17/21	5 0 1 T 5 B 0 0 9
	5 3 6		5 3 6 5 B 0 8 2
12/00	5 4 6	12/00	5 4 6 T
13/00	5 6 0	13/00	5 6 0 A

審査請求 未請求 請求項の数14 O L (全 29 頁)

(21) 出願番号 特願2000-158730(P2000-158730)
 (22) 出願日 平成12年5月29日(2000.5.29)
 (31) 優先権主張番号 特願平11-222559
 (32) 優先日 平成11年8月5日(1999.8.5)
 (33) 優先権主張国 日本 (J P)

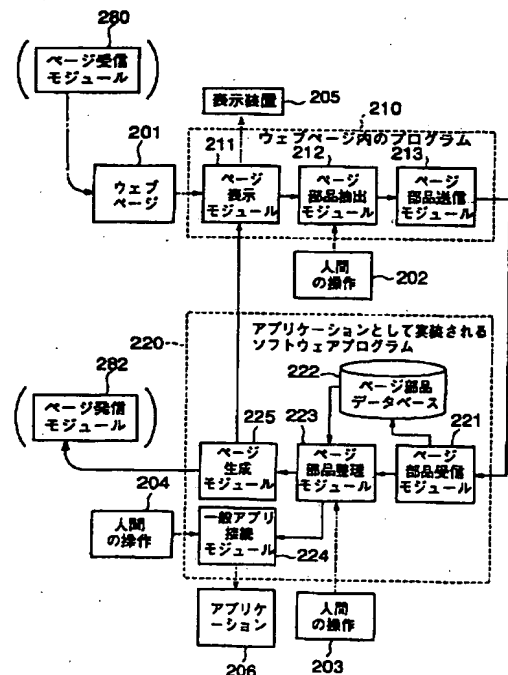
(71) 出願人 000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地
 (72) 発明者 浜田 伸一郎
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内
 (72) 発明者 關 俊文
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内
 (74) 代理人 100058479
弁理士 鈴江 武彦 (外6名)
 Fターム(参考) 5B009 NA04 NA06 NB01 VC01
5B082 AA01 EA07 GC04 GC06 HA06

(54) 【発明の名称】 ウェブページ部品統合処理方法及びクライアント装置

(57) 【要約】

【課題】 エンドユーザ側でウェブページの再構築等を行うことを可能とするウェブページ部品統合処理方法を提供すること。

【解決手段】 所望のウェブサーバから取得された所望のウェブページ201について、該ウェブページのデータ内に含まれる、該ウェブページから切り出して該ウェブページのその他の部分とは独立に処理可能なように記述された独立可能部分を特定するタグ情報を抽出し、このウェブページに含まれる独立可能部分がユーザ指定されたことが、抽出されたタグ情報に基づいて検出された場合、該ユーザ指定された独立可能部分の内容を含むページ部品を生成し(212)、生成されたページ部品を記憶し(222)、1または複数のウェブサーバから取得された複数のウェブページの各々から生成された複数のページ部品に対応する独立可能部分を含む新たなウェブページを生成する(223, 225)。



BEST AVAILABLE COPY

1

【特許請求の範囲】

【請求項1】 所望のウェブサーバから取得された所望のウェブページについて、該ウェブページのデータ内に含まれる、該ウェブページから切り出して該ウェブページのその他の部分とは独立に処理可能なように記述された独立可能部分を特定するタグ情報を抽出し、

このウェブページに含まれる独立可能部分がユーザ指定されたことが、抽出された前記タグ情報に基づいて検出された場合、該ユーザ指定された独立可能部分の内容を含むページ部品を生成し、

生成された前記ページ部品を記憶し、

1 または複数のウェブサーバから取得された複数のウェブページの各々から生成された複数のページ部品に対応する独立可能部分を含む新たなウェブページを生成することを特徴とするウェブページ部品統合処理方法。

【請求項2】 前記新たなウェブページのスタイルとして、該新たなウェブページに含ませたいずれの独立可能部分が元々含まれていたウェブページのスタイルを用いるか、または新たに定義されたスタイルを用いるかを、ユーザ選択に基づいて決定することを特徴とする請求項1に記載のウェブページ部品統合処理方法。

【請求項3】 前記独立可能部分に対するユーザ指定を受け付けるにあたって、対象となる前記ウェブページを、抽出された前記タグ情報に基づいて前記独立可能部分がユーザに識別可能なように表示することを特徴とする請求項1に記載のウェブページ部品統合処理方法。

【請求項4】 前記ページ部品は、少なくとも、対応する前記独立可能部分の内容と、該独立可能部分が含まれていたウェブページから全ての独立可能部分を取り除いた残りの部分の内容と、該独立可能部分が該ウェブページにおいて含まれていた位置に関する情報とを含むことを特徴とする請求項1に記載のウェブページ部品統合処理方法。

【請求項5】 前記独立可能部分のデータ内に、該独立可能部分に関連するアプリケーションプログラムに共するデータの所在を示す情報、該データのタイプを示す情報および必要な場合における該データのダウンロードに関する手続を含むアプリデータ属性情報が含まれている場合、該アプリデータ属性情報を前記ページ部品に含めて記憶することを特徴とする請求項1に記載のウェブページ部品統合処理方法。

【請求項6】 所望のウェブサーバから取得された所望のウェブページについて、該ウェブページのデータ内に含まれる、該ウェブページから切り出して該ウェブページのその他の部分とは独立に処理可能なように記述された独立可能部分を特定するタグ情報を抽出させ、

このウェブページに含まれる独立可能部分がユーザ指定されたことが、抽出された前記タグ情報に基づいて検出された場合、該ユーザ指定された独立可能部分の内容を含むページ部品を生成させ、

2

生成された前記ページ部品を記憶させ、

1 または複数のウェブサーバから取得された複数のウェブページの各々から生成された複数のページ部品に対応する独立可能部分を含む新たなウェブページを生成させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【請求項7】 所望のウェブサーバから取得された所望のウェブページについて、該ウェブページに含まれる予め定められた一纏まりの内容の部分に対するユーザ指定がなされたことが検出された場合、該一纏まりの内容の部分に対応するページ部品を生成するためのページ部品生成手段と、

生成された前記ページ部品を記憶するための記憶手段と、

複数の前記ページ部品を元にした新たなウェブページを生成するためのウェブページ生成手段とを備えたことを特徴とするクライアント装置。

【請求項8】 前記ウェブページはそのデータの中に前記予め定められた一纏まりの内容の部分に対応するデータを特定するタグ情報が記述されて前記ウェブサーバから送信されるものであり、

前記ページ部品生成手段は、前記タグ情報に基づいて前記ウェブページのデータの中からユーザ指定された前記一纏まりの内容の部分に対応するデータを取り出し、該データをもとに前記ページ部品を生成することを特徴とする請求項7に記載のクライアント装置。

【請求項9】 前記ウェブページは前記予め定められた一纏まりの内容の部分に対するユーザ指定がなされたことが検出されたときに該一纏まりの内容の部分に対応するデータを出力するプログラムが付加されて前記ウェブサーバから送信されるものであり、

前記ページ部品生成手段は、前記プログラムの動作によって出力された前記データをもとに前記ページ部品を生成することを特徴とする請求項7に記載のクライアント装置。

【請求項10】 前記ウェブサーバはショッピングサイトのウェブサーバであり、

前記ウェブページは前記ショッピングサイトのウェブサーバから取得されたショッピングページであり、

40 前記予め定められた一纏まりの内容の部分は前記ショッピングページに含まれる各々の商品カタログの部分であり、

前記新たなウェブページは複数のショッピングサイトのウェブサーバから取得された各々のショッピングページから選択された個々の商品カタログに対応するページ部品を元にして生成されたショッピングカートページであることを特徴とする請求項7に記載のクライアント装置。

【請求項11】 前記新たなウェブページを発信するためのウェブページ発信手段を更に備えたことを特徴とする

50

請求項7に記載のクライアント装置。

【請求項12】生成された前記新たなウェブページ内で、複数の前記ページ部品に含まれるデータに対し処理を行う手段を更に備えたことを特徴とする請求項7に記載のクライアント装置。

【請求項13】生成された前記新たなウェブページ単位もしくは前記ページ部品単位で、該新たなウェブページもしくは該ページ部品に対して行われた操作を保存する手段を更に備えたことを特徴とする請求項7に記載のクライアント装置。

【請求項14】所望のウェブサーバから取得された所望のウェブページについて、該ウェブページに含まれる予め定められた一纏まりの内容の部分に対するユーザ指定がなされたことが検出された場合、該一纏まりの内容の部分に対応するページ部品を生成させ、生成された前記ページ部品を記憶させ、複数の前記ページ部品を元にした新たなウェブページを生成させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、所望のウェブサーバが提供する所望のウェブページ上の所望の情報を統合的に処理するためのウェブページ部品統合処理方法及びクライアント装置に関する。

【0002】

【従来の技術】検索機能を提供する検索ページサイトのいくつかは、検索した結果、確からしいウェブページのURLをハイパーリンクとして列挙表示するウェブページを、自動的に生成する、といった技術を用いている。

【0003】また、電子商取引のサービスを提供する電子モールサイトのいくつかは、他の店舗のウェブページを取得して、ウェブページに含まれる情報を解析し、サイト内部のデータベースに格納する。そして、ウェブクライアントから要求があると、データベースから情報を引き出して、ウェブページを自動生成する。ただし、他の店舗のウェブページの記述形式は店舗によって違うため、電子モールサイトは、店舗ごとに異なる解析プログラムを用意するという高いコストを支払う方法を取っている。一方、そのような高いコストを避けるため、ウェブページの記述形式を全く統一するという試みがあるが、記述形式を全く統一することは、ウェブページの作者に強い制限を与えて、ページ記述の拡張性・自由度を失わせるという欠点を持つ。

【0004】

【発明が解決しようとする課題】まず、WWWにおいて、異なるウェブサーバや異なるウェブページに記載される情報を集めて分解・再構築して1つのウェブページに統合する際に、二つの問題点が存在する。1つは、集める対象となるウェブページごとに記述形式が異なると

いうことが原因となって、ウェブページの分解・解析に高いコストを要する点である。もう1つは、このコストを防ぐために集める対象となるウェブページの記述方法に強い制約を与えると、そのためにそれらウェブページの記述自由度が大きく引き下げられる点である。

【0005】また、WWWにおいて、異なるウェブサーバやウェブページに記載される情報を集めて分解・再構築して1つのウェブページにまとめる機能は、従来、ウェブサーバサイトにおいてのみしか行うことができなかった。したがって、ウェブページを閲覧するエンドユーザは、ウェブサーバサイトが用意した再構築結果しか構築できないという欠点があった。

【0006】本発明は、上記事情を考慮してなされたもので、ウェブページの記述方法に強い制約を与えることなく、ウェブページの分解・解析を低コスト化することを可能とし、また、エンドユーザ側でウェブページの再構築等を行うことを可能とするウェブページ部品統合処理方法及びクライアント装置を提供することを目的とする。

20 【0007】

【課題を解決するための手段】本発明（請求項1）に係るウェブページ部品統合処理方法は、所望のウェブサーバから取得された所望のウェブページについて、該ウェブページのデータ内に含まれる、該ウェブページから切り出して該ウェブページのその他の部分とは独立に処理可能なように記述された独立可能部分を特定するタグ情報（例えば、XMLの新しいタグでもよいし、HTMLの既存のタグに新しい意味付けを与えたものでもよい）を抽出し、このウェブページに含まれる独立可能部分がユーザ指定されたことが、抽出された前記タグ情報に基づいて検出された場合、該ユーザ指定された独立可能部分の内容を含むページ部品を生成し、生成された前記ページ部品を記憶し、1または複数のウェブサーバから取得された複数のウェブページの各々から生成された複数のページ部品に対応する独立可能部分を含む新たなウェブページを生成することを特徴とする。

30 【0008】本発明（請求項7）に係るクライアント装置は、所望のウェブサーバから取得された所望のウェブページについて、該ウェブページに含まれる予め定められた一纏まりの内容の部分に対するユーザ指定がなされたことが検出された場合、該一纏まりの内容の部分に対応するページ部品を生成するためのページ部品生成手段と、生成された前記ページ部品を記憶するための記憶手段と、複数の前記ページ部品を元にした新たなウェブページを生成するためのウェブページ生成手段とを備えたことを特徴とする。

40 【0009】好ましくは、生成された前記新たなウェブページ内で、複数の前記ページ部品に含まれるデータに対し（比較、合計金額見積もり、個数設定、一括発注等の）処理を行う手段を更に備えるようにしてもよい。

50

5

【0010】好ましくは、生成された前記新たなウェブページ単位もしくは前記ページ部品単位で、該新たなウェブページもしくは該ページ部品に対して行われた操作を保存する手段を更に備えるようにしてもよい。

【0011】なお、装置に係る本発明は方法に係る発明としても成立し、方法に係る本発明は装置に係る発明としても成立する。

【0012】また、装置または方法に係る本発明は、コンピュータに当該発明に相当する手順を実行させるための（あるいはコンピュータを当該発明に相当する手段として機能させるための、あるいはコンピュータに当該発明に相当する機能を実現させるための）プログラムを記録したコンピュータ読取り可能な記録媒体としても成立する。

【0013】本発明によれば、（エンド）ユーザは、任意のウェブサイトから取得したウェブページから選択した部分（部分情報）同士を組み合わせ、新しいページを作成することができる。このページは、その（エンド）ユーザだけのオリジナルであり、他人に公開しないで秘密にすることもできる。あるいは、他人に提供して使用して貰うこともできる。また、本発明によれば、ウェブページに対して、部分的に取り出せる領域を示す印（タグ情報）を付けるという記法によって、あるいは該部分に対応するデータを出力するプログラム（例えば、ドラッグイベントハンドラ）を付加することによって、情報を収集する手続きが汎用的になり、収集作業を低コスト化することができる。その一方で、ウェブページの作者自身が、部分的に取り出して欲しい領域の単位を収集する側に主張することができる。また、本発明によれば、上記印もしくは上記プログラムを付加することまたはその他の必要な情報を付加することなどのページ記述のルールに準拠すればよいから、ウェブページの作者に対して強い制限を与えないで済むという利点がある。

【0014】

【発明の実施の形態】以下、図面を参照しながら発明の実施の形態を説明する。

【0015】（第1の実施形態）図1は、本発明を適用するネットワーク環境の一例を示す。この環境下において、インターネットに代表されるコンピュータネットワーク基盤103を通じて、あるウェブクライアントサイト（ウェブクライアント側LAN）102におけるウェブクライアント（クライアント装置）101は、複数のウェブサーバサイト（ウェブサーバ側LAN）104～106におけるウェブサーバ（サーバ装置）107～109に存在する所望のウェブページを取得し、CRT等の表示装置（図示せず）を用いて表示する。なお、各サイトには複数のクライアント装置やサーバ装置が接続されても構わない。また、コンピュータネットワーク基盤103には、複数のウェブクライアントサイトが接続されるのが通常であるが、図1ではある1つのウェブクラ

6

イアントサイトのみ示してある。

【0016】本発明を適用したソフトウェアは、このような環境において、「ウェブクライアント101」および取得対象となる各々の「ウェブページ」の内部に配置される。

【0017】図2は、本実施形態のクライアント装置（計算機）上で実現されるシステムの全体構成例を示している。

【0018】本システムは、ウェブページ201および人間の操作（以下、ユーザ操作）202、203、204を入力対象、CRTあるいは液晶等の表示装置205およびアプリケーションプログラム206を出力対象とする。

【0019】また、本システムは、大きく二つのサブシステムを持っている。一つはウェブページ内に組み込まれる第1のソフトウェアプログラム210であり、もう一つはOS上で直接動作する一般的なアプリケーションとして実行される第2のソフトウェアプログラム220である。

【0020】本システムは、それらに加えて、それらを支援するページ受信モジュール280およびまたはページ発信モジュール282を備えてもよい。ページ受信モジュール280は外部システムからウェブページを受け取るためのモジュールであり、ページ発信モジュール282は外部システムへウェブページを発信するためのモジュールである。

【0021】ページ受信モジュール280は、外部システムからウェブページを取得して、第1のソフトウェアプログラム210に転送する。このページ受信モジュール280は、ウェブページをHTTPを介して取得することができることに加えて、POP3などのメールプロトコルを通じて取得することもできる。また、外部のページ発信モジュールとPoint-to-Pointで接続して、インスタントメッセージとして受け取ることもできる。メールプロトコルの場合、本文にウェブページが記載された形式のメールを受信することになる。メールプロトコルは、HTTPと違い、特定の人間から特定の人間へ転送することができる特徴があり、場合によってこれが利点となる。いずれのプロトコルを経由したとしても、ただちに第1のソフトウェアプログラム210に転送される。

【0022】図2に示されるように、第1のソフトウェアプログラム210は、ページ表示モジュール211、ページ部品抽出モジュール212、ページ部品送信モジュール213を含む。また、第2のソフトウェアプログラム220は、ページ部品受信モジュール221、ページ部品データベース222、ページ部品整理モジュール223、一般アプリ接続モジュール224、ページ生成モジュール225を含む。

【0023】さて、ウェブページ201が本システムに

入力されると、そのウェブページ内に組み込まれているソフトウェアプログラム210が駆動される。まず、ページ表示モジュール211が駆動されて、ウェブページは人間（以下、ユーザ）に読めるように整形される。その整形されたウェブページが表示装置205に出力されることによって、ユーザに対してウェブページが表示される。次に、ページ表示モジュール211は、獲得したウェブページ201を、ページ部品抽出モジュール212に送信する。

【0024】ページ部品抽出モジュール212は、ページ表示モジュール211から入力されるウェブページ201とユーザ操作202を両方受け取って初めて起動される。ページ部品抽出モジュール212は、起動されると、受け取ったウェブページから、ユーザ操作202により指定される部分文書を判定して（例えば、ウェブページが表示された表示画面上においてマウスカーソルにより指示された箇所に該当する部分文書を求めて）、その部分文書をもとにページ部品を生成する。さらに引き続いて、ページ部品送信モジュール213へ生成されたページ部品を送信する。

【0025】ここで、「部分文書」とは、ウェブページのデータ内に含まれる、該ウェブページから切り出して該ウェブページのその他の部分とは独立に処理可能なように記述された部分である。

【0026】部分文書は、例えば、これを特定するタグ（例えば、XMLの新しいタグでも、HTMLの既存のタグに新しい意味付けを与えたものでもよい）で特定される。なお、ウェブページと部分文書の具体例を後に示す（図4参照）。

【0027】また、「ページ部品」とは、ウェブページの部分文書を含み、他のページ部品と組み合わせられて新しいウェブページを構成することのできるデータ単位である。なお、ページ部品のデータ構造の具体例を後に示す（図9参照）。

【0028】ページ部品送信モジュール213は、ページ部品抽出モジュール212からページ部品を受信すると、ページ部品受信モジュール221に対して、ページ部品を送信する。

【0029】第2のソフトウェアプログラム220側のページ部品受信モジュール221は、ページ部品を受信すると、該ページ部品をページ部品データベース222に格納する。

【0030】ページ部品整理モジュール223は、ページ部品格納モジュール222に蓄積されているページ部品群に対して、検索や削除など種々のコマンドを実行することのできるモジュールである。ページ部品整理モジュール223は、ユーザ操作203により発行されるコマンドに従い、それらのコマンドを実行する。

【0031】一般アプリ接続モジュール224は、ページ部品整理モジュール223が持つコマンドの1つによ

って起動され、ページ部品内部のアプリケーション用のデータを取り出し、そのデータに対応するアプリケーションに対して、そのデータを送信するのに必要な手続きをユーザと対話的に実行する。

【0032】ページ生成モジュール225は、ページ部品整理モジュール223が持つコマンドの1つによって起動される。ページ生成モジュール225は、ページ部品データベース222に蓄積されているページ部品の全てまたは選択された一部のものを表示する（含む）ウェブページを生成する。作成されたウェブページは、ページ表示モジュール211へ送信される。

【0033】なお、ウェブページを表示する際には、部分文書の部分については、それが部分文書であること（ページ部品にできること）をユーザに分かるように表示等してもよい。

【0034】また、ページ生成モジュール225によって作成されたウェブページを外部システムに対して発信する場合は、そのウェブページをページ発信モジュール282へ転送する。ページ発信モジュール282は、ウェブページを受け取るとHTTPまたはSMTPなどのメールプロトコルを通じて外部システムに当ウェブページを発信する。メールプロトコルの場合、本文にウェブページが記載された形式のメールを受信することになる。メールプロトコルは、HTTPと違い、特定の人間から特定の人間へ転送することができる特徴があり、場合によってこれが利点となる。

【0035】図3は、本実施形態のシステムの論理的な処理の流れの概要を示している。

【0036】本システムは、例えば、外部からウェブページ（309）がダウンロードされることにより起動される（手順302）。その後、ダウンロードされたウェブページを解釈して、該ウェブページをCRT等の表示装置に表示する手順303を行う。この手順302、303の機能は、一般的なウェブブラウザの機能を利用して実装されるものである。

【0037】次に、手順302、303で処理されたウェブページを解析し、該ウェブページに含まれる、ページ部品の素材となる1または複数の部分文書のうち、ユーザ操作により指定されたものを特定した上で、その部分文書を抽出し加工してページ部品として生成し、それらを格納する手順を行う（手順304）。

【0038】ここまでの手順はウェブページに含まれるソフトウェアプログラム210として実装されるのに対して、以降の作業は一般的なアプリケーションのソフトウェアプログラム220として実装される。

【0039】さて、続いて、手順304において生成されたページ部品を転送する手順を行う（手順305）。この手順305の機能は、「当該ウェブページ内のソフトウェアプログラム210が、当該ウェブページ内部に維持するページ部品を、一般的なアプリケーションとし

て実装されるソフトウェアプログラム220の持つページ部品用のデータベース222に、コピーすること」を実現するものであり、このコピー機能自体はOSの機能を利用して実現される。例えば、Microsoft社の製品であるWindowsシリーズのOS上で本システムを動作させる場合は、OSの機能の1つであるOLE転送を用いて実装することによっても実現可能である。

【0040】次に、アプリケーションとして実装されるソフトウェアプログラム220が内部に持つページ部品をユーザのために整理して見せる手順を行う(手順306)。この手順はGUIを用いてユーザと対話的に行われるものであり、フォルダ作成、検索、部品削除などのコマンドが用意されている。

【0041】また、手順306の整理手順を行っている状態において、GUI上のコマンドを介して、図面CADアプリケーションや表計算ソフトなどといった一般アプリケーションヘデータを転送する手順307を動作させることができる。このアプリケーションヘデータを転送する手順307では、ページ部品を解析することにより、ページ部品にあらかじめ含まれているまたは関連付けられている、CADや表計算ソフトウェアといった様々なアプリケーション用のデータを取り出す作業に引き続き、取り出されたそれらアプリケーション用のデータを、そのデータに対応するアプリケーションヘ転送する手順が実行される。なお、その手順を行う前に、対応するアプリケーションがOS上で起動されていない場合は、その対応するアプリケーションをあらかじめ起動する。

【0042】また、手順306の整理手順を行っている状態において、GUI上のコマンドを介して、現在格納しているページ部品群を1つのウェブページとして再構築する手順308を動作させることができる。具体的には、このページの生成手順308では、格納されている全てのまたはユーザによって指定されたページ部品群のページ部品の各々がその内部に持つHTMLやXMLなどに代表されるマークアップ言語をもとに、それらの変換、結合、加筆作業等を行うことにより、様々なウェブページから切り出され維持されているページ部品を表示として含むウェブページを新たに構築する。また、この作業過程において、構成するページ部品のうち、XSLやCSSなどに代表されるスタイルシートが含まれている場合、ユーザは当該新たに作成するウェブページに対して適用するスタイルシートを選択することができる。さらに、この作業が終わると、そのページを表示する作業が行われる。

【0043】なお、ウェブページは、ウェブサーバからネットワークを介して取得したものでも構わない。例えば、ウェブサーバからネットワークを介して取得した他のウェブクライアントから記録媒体により取得

したものであってもよい。また、ネットワーク環境の外部で作成されたものであってもよい。また、本システムによって再構築されたウェブページであってもよい。

【0044】図4は、本実施形態のシステムに入力されるウェブページの一例である。この例を用いて、ウェブページの構成、構成要素名、構成要素の役割等について説明する。

【0045】図4において、401は、本システムが処理できるウェブページの一例であり、この例ではXMLで記述されている。なお、本システムは、HTML等の他の記述言語で記述されているウェブページに対しても、以下に説明する動作原理において、同様に処理することができる。

【0046】部分文書(404, 412)は、ウェブページの作者がウェブページ401において独立に取り出されることを意図しているウェブページの部分的な情報資源であり、ウェブページ401において1つ以上存在する。なお、ウェブページには部分文書を含まないものも存在し得るが、本システムは、1つ以上の部分文書を含むウェブページを対象とする。

【0047】また、1つの部分文書に対して1つの部分文書のルートタグが存在し、部分文書のルートタグ406および411は、それぞれ部分文書404および412の記述範囲を示す。具体的には、XMLやHTMLなどのマークアップ言語は、木構造をなしており、部分文書の記述範囲は、木構造上、指定されたルートタグ以下となる領域をもって示されている。

【0048】部分文書のルートタグ(406, 411)は、特定の印によって識別される。この印は、ウェブページ上のプログラム210により異なる。図4の例では、タグのcomponent属性を印としている。

【0049】また、部分文書(404, 412)は、アプリデータ属性情報(407)を1つ以上含むことができる。アプリデータ属性情報407は、部分文書404に関連付けられるアプリケーション用のデータのタイプ、所在、ダウンロードに必要な手続などを含む構造的な情報である。アプリデータ属性情報の記述範囲は、部分文書の記述範囲指定方法と同様、特定のタグに印をつけることにより示される。図4の例では、タグのappdata属性を印としている。

【0050】また、アプリデータ属性情報407は、アプリデータ取得手続き情報408、アプリデータの所在情報409、アプリデータタイプ情報410を含み、それぞれタグに印が付けられることによって記述位置が示される。図4の例では、それぞれタグ名procurl、data-url、app-typeを印としている。

【0051】アプリデータの所在情報409は、部分文書404が含むアプリケーション用のデータの所在を示すURLを持つ。本システムがユーザの入力コマンドに

したがって、アプリケーション用のデータをダウンロードする際には、このURLを参照する。また、このアプリケーション用のデータがオープンではなく、課金が必要であるとか会員に限定されているというように、限定的に公開されている場合がある。このような場合は、上記のダウンロードは失敗するので、データがダウンロードできるようにウェブページ上で手続きを行う必要がある。この手続きが用意されているウェブページのURLを示すものが、アプリデータ取得手続き情報408である。本システムは、ダウンロードには手続きが要求されていることを自動認識し、このアプリデータ取得手続き情報408をもとに、手続きが用意されているウェブページをユーザに開示する。また、アプリデータタイプ情報410は、ダウンロードの対象となるデータが、たとえば特定のCADアプリケーションや特定の表計算ソフトウェアなど、何のアプリケーション用のデータであるかを示す。この情報をもとに、本システムは、アプリケーション用のデータのダウンロードが成功したときに、自動的に対応するアプリケーションへそのデータを送信する。

【0052】図5は、ページ部品抽出モジュール212によるページ部品抽出手順(図3の304)の詳細な論理的手順の一例を示している。

【0053】ページ部品抽出手順304は、ウェブページとユーザ操作にしたがってページ部品を作成する。具体的には、まず、ページ部品の「内容部」を作成する

(手順501)。手順501については、図6にて詳細に説明する。次に、ページ部品の「管理情報部」を作成する(手順502)。手順502については、図7にて詳細に説明する。次に、ページ部品の「アプリデータ属性情報部」を作成する(手順503)。手順503については、図8にて詳細に説明する。最後に、手順501で作成されたページ部品の内容部と、手順502で作成されたページ部品の管理情報部と、手順503で作成されたページ部品のアプリデータ属性情報部とを合わせることによって、ページ部品を完成する(手順504)。

【0054】なお、ページ部品は部分文書毎に作成するので、1つのウェブページからN個の部分文書が指定された場合には、ページ部品抽出手順が部分文書毎に繰り返されて、N個のページ部品が作成される。この場合に、重複する不要な処理が存在する場合には、これを検出して省くようにしてもよい。

【0055】図6は、図5における手順のうちのページ部品の内容部を作成する手順501の詳細な論理的手順の一例を示している。

【0056】ページ部品内容抽出モジュール610は、ページ部品の内容部を作成する手順501を実行するソフトウェアプログラムモジュールである。ページ部品内容抽出モジュール610は、ユーザ操作601とウェブページ602を入力とし、内容部603を出力とする。

【0057】まず、ページ部品内容抽出モジュール610は、ウェブページ内でユーザ操作601により指定された部分文書を特定する(手順611)。ユーザ操作601は、ウェブブラウザにおけるページ表示を直接利用して、部分文書を指定するものであり、具体的には、一般的なウェブブラウザにおいて表示されたウェブページ上の領域で、たとえばドラッグ操作やクリック操作やコンテキストメニュー実行などの操作を行うことにより、部分文書を指定することができる。本実施形態では、ウェブブラウザがサポートするHTMLなどのマークアップ言語のウェブページに関するイベント配送機構を利用して、このようなウェブページ上の操作を実現するものとする。

【0058】次に、ページ部品内容抽出モジュール610は、特定されたルートタグを接続位置情報621として保存する(手順612)。

【0059】次に、ページ部品内容抽出モジュール610は、特定された部分文書のルートタグ以下の文書を内容文書622として保存する(手順613)。

【0060】次に、ページ部品内容抽出モジュール610は、ページ内部に含まれるすべての部分文書をページから除去したものをコンテナ文書623として保存する(手順614)。

【0061】次に、手順612~614で得られた接続位置情報・内容文書・コンテナ文書を構成して、当該ページ部品の内容部603を作成する(手順615)。

【0062】図7は、図5における手順のうちのページ部品の管理情報部を作成する手順502の詳細な論理的手順の一例を示している。

【0063】ページ部品管理情報生成モジュール710は、ページ部品の管理情報部を生成する手順502を実行するモジュールである。ページ部品管理情報生成モジュール710は、ページ部品の管理情報部701を出力とする。

【0064】具体的な論理的手順を以下に説明する。

【0065】まず、対象となるウェブページに関して、補助的な外部モジュールが存在するかどうかを判断する(手順711)。ウェブページを補助する外部モジュールとしては、たとえばイメージファイル、音楽ファイル、文書型ファイル、スタイルシートファイル、スクリプトファイルなどがある。ウェブページがXMLの場合、外部モジュール名が記述されている領域のタグに印を付けるというルールを課すことにより、処理手続きが外部モジュール名を取り出すことができ、これにより判断を下す。図4の例402および405では、タグにexternal属性が添付されていることを印としている。外部モジュールが存在する場合は手順712へ移行し、存在しない場合は手順718へ移行する。

【0066】次に、外部モジュールが存在する場合、存在するすべての外部モジュールをクライアントサイトの

13

ファイルシステムにコピーする(手順712)。

【0067】このコピーに際しては、クライアントサイトのコピー先ディレクトリは、ウェブサーバサイトのディレクトリの相対位置を再現するように選択する。すなわち、ミラーリングする。このようにコピーすることにより、外部モジュールがウェブページ内部で相対的に参照指定されていたとしても、図3の手順308においてウェブページを再構成する際に、特別に参照位置を書き直す処理する必要がないという利点がある。ミラーリングの一例として、コピー元のURLのうち、ホスト名部をディレクトリ名に変更しコピー先のディレクトリ名に変更してコピー先のディレクトリとする方法がある。例えば、コピー元のURLが、

`http://www.domain.comain.com/dirl/dir2/file1`

の場合、`file1`を、

`/www.domain.com/dirl/dir2`

にコピーする。この方法だと、複数のウェブサーバサイトから外部モジュールをコピーしても、ファイル名が重なることによって上書きされることを未然に防止することができる。

【0068】手順712に引き続き、これらミラーリングされた外部モジュールに関して、コピー先のディレクトリをフルパスで添付した形式で、それら外部モジュール名を管理情報部701内部に追加する(手順713)。

【0069】手順713に引き続き、外部モジュールの中にスタイルシートが存在するかどうかを判断する(手順714)。ウェブページがXMLの場合、スタイルシート名が記述されている領域のタグに印を付けるというルールを課すことにより、処理手続きがスタイルシート名を取り出すことができ、これにより判断を下す。スタイルシートが存在する場合は手順715へ移行し、存在しない場合は手順716へ移行する。

【0070】手順715では、手順712でミラーリングされたスタイルシートに関して、コピー先のディレクトリをフルパス形式で、それらスタイルシート名を管理情報部701内部に追加する。結果として、手順713で書きこまれた外部ファイル名と重複する。

【0071】手順716では、外部モジュールの中に文書型定義ファイルは存在するかどうかを判断する。ウェブページがXMLの場合、文書型定義ファイル名が記述されている領域のタグに印を付けるというルールを課すことにより、処理手続きが文書型定義ファイルを取り出すことができ、これにより判断を下す。もしくはDOM(Document Object Model)という標準APIを用いることにより、取り出すこともできる。文書型定義ファイルが存在する場合は手順717へ移行し、存在しない場合は手順718へ移行する。

【0072】手順717では、手順712でミラーリングされたスタイルシートに関して、コピー先のディレク

14

トリをフルパス形式で、これらスタイルシート名を管理情報部701内部に追加する。結果として、手順713で書きこまれた外部ファイル名と重複する。

【0073】手順718では、ウェブページのURLを管理情報部701内部に追加する。

【0074】以上の手順で完成された管理情報部701をもって、ページ部品管理情報生成モジュール710の出力結果とする。

【0075】図8は、図5における手順のうちのページ部品の管理情報部を作成する手順503の詳細な論理的手順の一例を示している。

【0076】ページ部品アプリデータ属性情報生成モジュール810は、ページ部品の管理情報部を生成する手順503を実行するソフトウェアプログラムモジュールである。ページ部品管理情報生成モジュール810は、ページ部品のアプリデータ属性情報部801を出力とする。

【0077】具体的な論理的手順を以下に説明する。

【0078】まず、対象とする部分文書に含まれるアプリデータ属性情報をすべて特定する(手順811)。この特定は、XMLの場合、アプリデータ属性情報に付けられた印を元に部分文書内を検索することで実現する。図4の例407では、タグに`appdata`属性が添付されていることを印としている。

【0079】引き続き、特定されたアプリデータ属性情報すべてを、アプリデータ属性情報部801として格納する(手順812)。

【0080】以上の手順で完成されたアプリデータ属性情報部801をもって、ページ部品アプリデータ属性情報生成モジュール810の出力結果とする。

【0081】図9は、図5から図8で説明したページ部品作成手順において扱うページ部品のデータ構造の概要をまとめたものである。

【0082】ページ部品900は、ウェブページの再構成およびアプリケーションへのデータ転送を実現するために必要な情報を維持している。すなわち、図9に示されるように、ページ部品900は、内容部910、管理情報部930、アプリデータ属性情報部920を含む。

【0083】内容部910は、ページ部品が持つウェブページの部分文書を組み立てる際に必要な情報を維持する。内容文書911は、ウェブページから切り出された部分文書そのものが格納されている。これはウェブページの再構成においてそのまま再利用される。コンテンツ文書912は、ウェブページにおける部分文書(404や412など)以外の領域の文書が格納されている。HTMLやXMLなどのマークアップ言語は、ルートタグが一つであることがルールとなっているため、複数のページ部品900の内容文書を並べてつなげるだけでは、有効な文書として再構成することはできない。このためウェブページとして構成するためには、複数の部分文書に

加えてそれらをサブツリーとして包含する文書が必要となる。その文書がコンテナ文書912である。接続位置情報913は、コンテナ文書912内において複数の部分文書(404や412など)をサブツリーとして挿入できるノードを示している。

【0084】管理情報部930は、組み立てられたウェブページを管理するのに必要な情報を維持する。URL931は、ページ部品の取りだし元のURLを維持する。これは、手順712でのミラーリングやページ部品の整理や手順1212におけるスタイルシートの種類の識別などに利用される。外部モジュール名リスト940は、クライアントサイトのファイルシステムにミラーリングされた、ウェブページを表示可能とするのに必要な外部モジュール名すべてがフルパス形式で格納されている。ページを表示可能とするのに必要な外部モジュール名すべてがフルパス形式で格納されている。外部モジュール名は、スタイルシートファイル名941と文書型定義ファイル名942とその他の外部モジュール名943の3種類に区別されて維持される。

【0085】アプリデータ属性情報部920は、ページ部品900が持つアプリケーション用のデータに関する情報を維持する。具体的には、アプリデータ属性情報921を複数維持する。

【0086】図10は、図3におけるページ部品の整理手順306の詳細な論理的手順の一例を示している。

【0087】ページ部品整理モジュール223は、今までに獲得した複数のページ部品を整理し、ユーザが発行するコマンド群にしたがって、さまざまな機能を提供するモジュールである。

【0088】ページ部品整理モジュール223は、今までにウェブページから集めてきたページ部品項目を列挙して表示する、部品項目表示モジュール1010を含んでいる。

【0089】部品項目表示モジュール1010は、ページ部品の項目を列挙して表示する方式として3種類の方式を持っており、これらは表示モードを切り替えるコマンドによって、表示の方式を選択することができる(手順1011)。また、いずれの表示方式においても、表示されているページ部品の項目を選択することができ、選択されている項目はフォーカスされる。以下に説明するコマンドのうち、ページ部品に対して働きかけるコマンドにおいて、維持しているページ部品すべてに適用されるものと、単一のページ部品に適用されるものがあり、後者の場合、フォーカスされているページ部品に対してコマンドは適用される。

【0090】表示の仕方は、

- ・分類なしの表示方式1012、
- ・ユーザがフォルダーを動的に作成して分類することのできる表示方式1013、
- ・部品内部にあらかじめ含まれているクラス情報を元に

自動的にフォルダーを作り分類表示する方式1014、の3種類が用意されている。

【0091】ページ部品整理モジュール223は、部品単一閲覧モジュール1020を含んでいる。

【0092】部品単一閲覧モジュール1020は、部品項目表示モジュール1010においてフォーカスされているページ部品を単一で、2種類の閲覧モードで表示する機能を有し、コマンド1021によって切り替えることができる。具体的には、閲覧モードは、

- ・対象となるページ部品がコピー元のウェブページにおいて適用されていたスタイルで表示するオリジナルスタイル再現モード1022、
- ・部品単一閲覧モジュール1020があらかじめ持っているスタイルで表示する統一スタイル生成モード1023、

の2種類が用意されている。

【0093】オリジナルスタイルを再現する手順1022では、ページ部品の内容部を元に、ローカルファイルシステム上にウェブページのソースファイルを生成する。具体的には、内容部のコンテナ文書912に対して、接続位置913において内容文書911を挿入するという作業によりソースファイルの内容を得る。ページ部品の内容部がXMLの場合、さらにそのソースファイル内にスタイルファイルへの関連付けの記述を加筆する。それらの手順において作成されたウェブページのソースファイルを表示する。

【0094】一方、統一スタイルで閲覧する手順1023では、ページ部品の内容部がXMLおよびそれに準ずるマークアップ言語で記述されているときのみ有効であり、次の手順で実現される。まず、ページ部品の内容部をもとに、ローカルファイルシステム上にウェブページのソースファイルを生成する。さらに、ソースファイル内に、部品単一閲覧モジュール1020があらかじめ用意したスタイルファイルへの関連付けの記述を加筆する。それらの手順において作成されたウェブページのソースファイルを表示する。

【0095】また、部品項目表示モジュール1010において列挙されているページ部品群に関して、4種類のコマンド(1002~1005)を適用することができる(手順1001)。

- ・ページ生成コマンド(1002)を発行すると、維持している全部または一部のページ部品を含むウェブページが生成され、一般的なウェブブラウザで表示される。
- ・アプリデータ転送コマンド(1003)を発行すると、フォーカスされているページ部品が含んでいるアプリケーション用のデータをそれに対応するローカルアプリケーションに転送するのに必要な手続きを対話的に実行する。

・検索コマンド(1004)を発行すると、維持しているページ部品の中から、タグ名(すなわち要素名)とそ

10

20

30

40

50

17

の内容（すなわち要素値）を元に条件を記述し、それにマッチするページ部品を見つけ出すことができる。

・削除コマンド（1005）を発行すると、フォーカスされているページ部品が削除される。

【0096】図11は、図3における一般アプリケーションへの接続307の詳細な論理的手順の一例を示している。

【0097】一般アプリ接続モジュール224は、一般アプリケーションへの接続を行うモジュールであり、具体的にはページ部品が含まれているアプリケーション用のデータをそれに対応するローカルアプリケーションに転送するのに必要な手続きを対話的に実行する。

【0098】まず、手順1101において、ページ部品に含まれているアプリデータ属性情報407をすべて抽出する。

【0099】その後、手順1102において、ページ部品に含まれているアプリデータ属性情報407が1セットであるかどうかを判定し、複数セットある場合は手順1104へ移行し、1セットしかない場合は手順1105へ移行する。

【0100】手順1104では、選択メニューコマンドなどのユーザ操作を受け取り、複数存在するアプリデータ属性情報407のうち、今回取り出す対象とするアプリケーション用のデータを決定する。

【0101】手順1105では、アプリケーション用のデータをダウンロードする際に必要な手続きが存在するかどうかを判定する。具体的には、アプリデータ属性情報を解析しアプリデータ属性情報407に、アプリデータ取得手続き情報408が含まれている場合は手続きが存在すると判断し、含まれていない場合は手続きが存在しないと判断する。手続きが存在すると判断された場合は手順1106へ移行し、手続きが存在しないと判断された場合は手順1107へ移行する。

【0102】手順1106では、手続きに必要なウェブページを開き、ユーザに手続きを行わせる。具体的には、ウィンドウを開き、そのウィンドウ上にアプリデータ属性情報407に含まれているアプリデータ取得手続き情報408に記述されているURLのウェブページを表示して、ユーザに手続きを行わせる。

【0103】手順1107では、アプリデータ属性情報に含まれているアプリデータの所在情報409に記述されているURLからアプリケーション用のデータをウェブサーバからダウンロードする。引き続き、手順1107では、アプリデータ属性情報407に含まれているアプリデータタイプ情報410に記述されている識別情報から、対象のアプリケーション用のデータに対応するアプリケーションを特定する。

【0104】そして、手順1109では、対応するアプリケーションに対してダウンロードされたアプリケーション用のデータを転送する。ただし、対応するアプリケ

18

ーションが起動されていない場合は、アプリケーション用のデータを転送する前に、対応するアプリケーションを起動しておく。本実施形態では、この手順1109について、OSのシェルの機能を利用して実現するものとする。

【0105】図12は、図3におけるページの生成308の詳細な論理的手順の一例を示している。

【0106】ページ生成モジュール225は、維持している全部または一部のページ部品を含むウェブページを生成するモジュールである。

【0107】まず、手順1211において、ページ部品データベース222から対象となるページ部品を取り出す。

【0108】次に、手順1212において、それぞれのページ部品における管理情報部のスタイルシートファイル名941を参照することにより、ページ部品に含まれるスタイルシートをリストアップする。このとき、管理情報部のスタイルシートファイル名が一致している場合は同一スタイルシートであるとみなし、重複するスタイルシートは代表を一つ残して、他のスタイルシートはリストアップしない。

【0109】続いて、手順1213では、手順1212においてリストアップされたスタイルシートの適用されたイメージを、ダイアログボックスに列挙し、ユーザにスタイルシート1204を選択させる。ここまでするウェブページに適用するスタイルシートの選択するための手順である。

【0110】次に、手順1214において、対象となるページ部品群1202の中からページ部品を一つ取りだし、その内容部910のコンテナ文書912をページ草案1205として一時的に維持する。

【0111】次に、手順1215において、取り出したページ部品の内容部910の接続位置情報913を参照することにより、接続位置に対して内容文書911をページ草案1205に挿入する。

【0112】続いて、手順1216において、同じ接続位置に対して、その他のページ部品の内容文書911を挿入する。

【0113】最後に、手順1217において、手順1213にて選択されたスタイルシートを関連付ける記述をページ草案に対して加筆し、ウェブページを完成させる。

【0114】以下では、実行画面例に沿って本システムの利用手順を詳細に説明する。おおまかな手順は以下の通りである。

(1) ウェブページページから本システムへのページ部品のピックアップ

(2) 集めたページ部品の整理

(3) 集めたページ部品の検索

(4) ページ部品群から新規ウェブページの作成

図13は、実行画面例の一つを表している。

【0115】1300は一般的なウェブブラウザ（に対応するウィンドウ）であり、ウェブページを表示する領域を持っている。

【0116】1320は、アプリケーションとして実装される、図2の第1のソフトウェアプログラム220に対応する部分である。

【0117】1321は、ページ部品整理モジュール223の部品項目表示モジュール1010に相当し、今までに蓄積したページ部品群（1323）が表示される。

【0118】1322は、部品単一閲覧モジュール1020に相当し、部品項目表示モジュール1010に表示されているページ部品群のうち、フォーカスされているページ部品の内容が表示される。

【0119】ウェブページ1310は、複数の部分文書を含み、ページ内部に図2の第1のソフトウェアプログラム210に相当するプログラムを内蔵している。図13の例では、ウェブページ1310は、3つの部分文書（1311～1313）が画面内に列挙して表示されており、また、各部分文書には画像（1315～1317）が含まれている。

【0120】この部分文書の内容はどのようなものであってもよい。例えば、ウェブページが音楽CD、書籍、工業部品等の商品のカタログページ、部分文書が個々の商品毎の情報、画像がその商品の外形図である。あるいは、ウェブページがニュースページ、部分文書が1つのニュース、画像がそのニュース写真である。あるいは、ウェブページが検索サービスページ、部分文書が1つの検索結果、画像がその検索結果に関連する映像である。

【0121】ウェブページ（1310）を表示する際には、部分文書（1311～1313）については、それが部分文書であること（ページ部品にできること）をユーザに分かるように表示等してもよい。例えば、部分文書であることを示す簡単な画像を部分文書に上書きして表示する方法、部分文書の輪郭を特定の色に着色する方法、これらの処理をマウスカーソルが部分文書上に位置するときのみ行う方法、マウスカーソルが部分文書上に位置するときカーソルの表示を変化（例えば、色変化、形状変化）して、もしくは音を発生して、もしくは発生している音を変化させて通知する方法、ドラッグできたことをもって部分文書であることを知らしめる方法など、種々の方法が考えられる。

【0122】なお、図13の例では、ウェブブラウザのウィンドウ内に本システムに対応する領域が表示されているが、ウェブブラウザのウィンドウと本システムに対応するウィンドウとを別々にしてももちろん構わない。

【0123】本システムによって、表示されたウェブページ内の部分文書（1311～1313）は、ページ部品として取り出すことができるように仕組まれており、例えばユーザが所望の部分文書（1311～1313）

の任意の部分（図13の例では画像（1315～1317）が表示された部分）をドラッグ（1331参照）することにより、ページ部品として取り出すことができる。このドラッグ操作は、ユーザ操作202にあたり、ページ部品抽出モジュール212およびページ部品送信モジュール213が順に起動される。そして、例えばドラッグ中のページ部品を部品項目表示モジュール1010用表示領域1321にドロップ（1332参照）することにより、ページ部品がページ部品受信モジュール221に渡され、ページ部品データベース222に格納される。

【0124】以上の操作を通じて、ユーザは所望の情報をウェブサーバサイトからピックアップし、ページ部品として収集することができる。

【0125】また、このページ部品収集作業は、ウェブページやサイトをを超えて行うことができる。例えば、まず、会社Aのカタログページにおいて、いくつかのカタログ（部分文書）を本システムにピックアップし、続いて、会社Bのカタログページへ立ち寄り、そのページからもいくつかのカタログ（部分文書）をピックアップするといったことが可能である。これにより、サイトを超えて情報を収集し比較検討することが可能となる。

【0126】部品単一閲覧モジュール1322は、スタイル形式およびジャケット形式の2種類の閲覧モードを持っており、それぞれオリジナルタイル再現機能1022および統一スタイル1023に相当する。

【0127】図14（a）に示すように、スタイル形式1401は、ページ部品がその内部に持っているスタイルシートを適用して表示する方式である。したがって、ページ部品を取得してきた元のサイトで表示されていたスタイルと同じとなる。

【0128】一方、図14（b）に示すように、ジャケット形式1402は、すべてのページ部品に対して適用できる共通のスタイルで表示するものである。

【0129】部品項目表示モジュール1321は、ページ部品を図15の1501に示されるユーザフォルダ方式、図16の1601に示される自動カテゴリ方式、図17（a）の1701（大きいリスト表示）あるいは図17（b）の1702（小さいリスト表示）に示されるリスト方式の3つの表示方式を有しており、これらは下部についているタブによって切り替えることができる。

【0130】ユーザフォルダ方式1501は、集めてきたページ部品をフォルダを用いて分離することができる方式である。ユーザは任意にフォルダを作成、削除することができ、任意フォルダ下に集めてきたページ部品を配置することができる。例えば、会社Aのウェブサーバサイトから集めてきたページ部品は会社Aフォルダの下に置くとし、会社Aフォルダ下に、「売却済み」「これから購入」などのサブフォルダを作成するといったことができる。

21

【0131】自動カテゴリ方式1601は、ページ部品内部にあらかじめ埋め込まれたカテゴリ情報を用いて、自動的にカテゴリ表示する方式である。図16において、1602はカテゴリを表し、1603はページ部品を表す。

【0132】例えば、PLIBでは工業部品の分類が業界において定義されているので、これを利用し例えばISO13584(PLIB)という工業部品規格に沿ってカタログを表示させることができる。その他の分野としては音楽CDや書籍などが考えられる。このように、業界側でトップダウンにアイテムの分類が定義されているような適用領域において、カテゴリ方式1601は有効である。

【0133】リスト方式1701、1702は、カテゴリズを行わない表示方式であり、とりあえず放り込む買い物カゴのイメージである。表示方式は、大きなアイコン表示1701と小さなアイコン表示1702の2種類があり、メニューによって切り替えることができる。

【0134】また、内部に格納しているページ部品群に対して条件を指定して検索を行うことができる。これにより格納されたページ部品の個数が非常に多くなっても、対象を簡単に見つけ出すことができる。この機能は検索コマンド(1004)に相当する。検索コマンドは編集メニューから発行する。

【0135】検索コマンドを発行すると、図18に例示するようなダイアログボックス1801が現れる。このダイアログボックス1801に検索条件を入力して検索開始ボタン1804を押すと、検索が実行される。検索条件の指定方法は、項目別にマッチさせたい値を入力する方法(1802)と、問い合わせ言語であるXQL形式で記述する方法(1803)の2種類ある。検索を実行した結果、検索にマッチしたページ部品が存在した場合は、部品項目表示モジュール1321において当該ページ部品がフォーカスされる。

【0136】本システムは、以上で説明した手順で複数のウェブサーバサイトからページ部品というモジュール単位で、情報を集めてくることができる。以降で説明する機能は、これら集めてきたページ部品群を一つのウェブページに再構成する機能である。これはページ生成モジュール225に相当する。

【0137】ページ部品群を集めたウェブページを作成するには、メニューからブラウズコマンドを発行する。すると、図19に例示したようなダイアログボックス1900が現れる。このダイアログボックスはこれから作成するウェブページに適用するスタイルを選択するものである。候補として挙げられているスタイルは、内部に格納しているページ部品群が保持しているスタイルである。重複している場合は代表して一つだけ表示される。

【0138】例えば、二つのウェブサイトから情報を取ってきている場合には、(それらが同じスタイルでない

22

とすると)選択する候補として2種類のスタイルが呈示される(1901)。なお、領域1902では、1901にてフォーカスされているスタイルのサンプルイメージを表示している。

【0139】また、ファイル指定ボタン1903を押すと、図20の2000のようにダイアログボックスが変化する。ここで現れたエディットボックス2001にスタイルファイル名を記述することにより、クライアントサイトのファイルシステムに存在するスタイルを適用することができる。これは、サーバサイトに存在するページに対して顧客の資源を適用することができることを意味している。例えば、外部からカタログなどの情報を集めてきて、それを自社のスタイルを適用してページを再構成するということが可能となる。

【0140】以上の手順により適用するスタイルを決定すると、内部に格納しているページ部品群を元に、それらページ部品の情報の全部または一部を含むウェブページが構築され、例えば図21に示すように、ウェブブラウザのページウィンドウに表示される。

【0141】なお、本実施形態では、本システムを実現するプログラムを、ウェブページ内に組み込まれる第1のソフトウェアプログラムと、OS上で直接動作する一般的なアプリケーションとして実装される第1のソフトウェアプログラムとに分割して構成したが、第1のソフトウェアプログラムをウェブページにかかわらず共用できるように共通化し、第1のソフトウェアプログラムと第2のソフトウェアプログラムをOS上で直接動作する一般的なアプリケーションとして実装される一体化プログラムとして実施することも可能である。

【0142】また、上記の第2のソフトウェアプログラムや一体化プログラムは、記録媒体に格納した形でもしくは通信媒体を利用して受け渡し等することができる。

【0143】また、本実施形態のウェブ文書内の部分文書の中にさらに部分文書を含ませるような階層構造を採用することも可能である。

【0144】(第2の実施形態)続いて、本発明をショッピングカートへ適用した場合の実施形態について説明する。

【0145】昨今、WWWベースの電子商取引システムあるいは電子店舗システム、特に個人客を対象としたB2Cショッピングサイトなどが、よく見かけられるようになった。ショッピングサイトの多くは、商品検索や比較、購入候補商品の合計金額の見積もり計算、一括した購入手続きなどの機能を提供し、顧客の便宜を図っている。

【0146】電子店舗を提供するショッピングサイトでは、ショッピングカート機能を提供するものもある。ショッピングカート機能では、通常、客は、その電子店舗で選んだ商品をショッピングカートに入れたり、一度入れた商品を自由にカートから取り出したりすることがで

10

20

30

40

50

き、最後に例えば「購入ボタン」あるいは「決定ボタン」あるいは「確定ボタン」を押すなどすることによって、そのときにショッピングカート内に入っている1または複数の商品についてまとめて購入手続きをすることができるようになっている（ここでクレジット番号を入力するなどして複数の商品について一括して支払い手続きを行うことも多い）。

【0147】このようなショッピングサイトは各々、閉じたシステムとして構築されており、いずれのショッピングサイトも他のショッピングサイトとの連携を取る機能を持っていない（なお、単一店舗を提供するショッピングサイトの他に、複数店舗を提供するショッピングモールのショッピングサイトも散見されるようになったが、サイトの運営が複数ベンダーの出資によって行われているという違いがあるだけであって、システム的には単一店舗のショッピングサイトと同様である）。

【0148】したがって、オンラインで商品を購入するに際して複数の購入先ショッピングサイトの候補がある場合、顧客は、候補となる各々のショッピングサイトにアクセスし、ページ切り替えや画面スクロールやページ・ジャンプなどを何度も繰り返し行いまた必要に応じてメモ・ソフトまたは現実の紙にメモを取るなどして、所望の商品を探し求め、商品の比較を行い、合計金額を計算するなどの作業を行う必要があり、そして、複数のショッピングサイトから商品を購入する段階になったときは該当する各々のショッピングサイトに逐一アクセスし直して発注手続きをしなければならない。このように、顧客は不便を強いられることになる。

【0149】そこで、本実施形態では、クライアント装置側において、各ショッピングページから選択した商品カタログをページ部品にしてクライアント装置が生成するウェブページ（ショッピングカートページ）へ挿入可能とすることによって、複数のショッピングサイトにまたがったショッピングカート機能をクライアント上で実現するようにしている。詳しくは後述するが本実施形態では、ウェブページからピックアップできる文書単位を（ショッピングサイトから提供されるウェブページに含まれる）商品カタログとし、商品カタログ群を取り扱うことのできるコンテナ文書およびスタイルシートをクライアント・システム内に用意し、これによって商品を列挙するウェブページ（ショッピングカートページ）を合成できるようにしている。さらに詳しくは後述するが本実施形態では、スタイルシートの中にクライアント側で実行される（例えば、JavaScript、VBScript、AppletあるいはActiveXなどによる）プログラムを埋め込み、このプログラムにウェブページに対する何らかのユーザ操作に反応したときあるいはウェブページの表示のときに実行が開始され、各商品のデータ項目（例えば商品の価格情報等）を参照し、必要な処理（例えば合計金額の計算あるいは個数の

変更等）を行い、その結果をウェブページの表示に反映させる（フィードバックする）機能を持たせることにより（後述する各機能エンジン）、種々の機能を提供できるようにする（なお、スタイルシートは、あらかじめシステム内に用意しておいてもよいし（内蔵スタイル）、どこかのショッピングサイトから得たものを使ってもよい）。

【0150】まず、本実施形態の概要について説明する。

【0151】本発明を適用するネットワーク環境の一例は図1で説明したものと同様である。本実施形態では、ウェブサーバサイトは電子店舗事業者のショッピングサイトであり、ウェブクライアントは電子店舗の顧客となるユーザの使用するコンピュータシステムである。

【0152】本実施形態では、顧客のウェブクライアント上に配置されたショッピングカートシステムを中心として、個々の独立したショッピングサイトを連携させる。

【0153】以下では、クライアント側のシステムを、「パーソナルショッピングモールシステム」と呼ぶものとする。

【0154】このパーソナルショッピングモールシステムは、次の機能を提供する。

（1）任意のショッピングサイトのウェブページすなわちショッピングページ（商品一覧を含むページ）から任意の商品の商品カタログをピックアップして、ウェブクライアント上に配置されたショッピングカートに格納する機能

（2）ウェブクライアント上のショッピングカートに格納された商品カタログ群に関して、ウェブクライアント上で比較、ソート、合計金額の見積もりなどの処理を行う機能

（3）ウェブクライアント上のショッピングカートに格納された商品群を購入するための手続きを一括して行うための機能

これにより、任意のショッピングサイトの任意の商品群に関する商品比較、ソート、合計金額の見積もり、一括購入等が実現される。

【0155】本実施形態では、ウェブクライアント上で上記機能を実現するために、ショッピングサイトの提供するショッピングページには、次のような条件が課せられるものとする。

・ショッピングページの各商品に関して、ウェブクライアント上から「商品データ」、「商品の表示スタイル」を読み取ることができるようになっている。ウェブクライアント上において、商品データに加えて商品の表示スタイルを必要とするのは、商品データをショッピングページ上からピックアップするためのイベントハンドラをセットするとき、その商品データの表示領域を知る必要があるからである。

25

・商品データには、「価格」および「発注手続き」が含まれている。

【0156】ショッピングサイトの提供するショッピングページに課せられるこのような必要条件を「モール参加規約」と呼ぶものとする。ショッピングサイト事業者は、そのショッピングサイトが提供するショッピングページをモール参加規約に適合させることで、パーソナルショッピングモールシステムに利用してもらうことができる。このようなモール参加規約は、上記必要条件を満たす様々なバージョン（バリエーション）が考えられる。

【0157】以下では、本実施形態についてより詳しく説明する。

【0158】本実施形態のパーソナルショッピングモールシステムでは、モール参加規約を次の内容としている。

(1) 各商品カタログの表示部に対してドラッグイベントハンドラを用意し、そのハンドラではXML形式の当該商品データ（部分文書）をクリップボードへコピーする処理を入れること。

(2) 商品データには「価格」が含まれていること。

(3) 商品データには「発注手続き」に関するフォームページへのURLが含まれていること。

上記のモール参加規約では、ショッピングページをXMLページまたはHTMLページのいずれで提供してもかまわない。

【0159】XMLページで提供する場合は、例えば、次のようなページ実装が考えられる。

(1) ショッピングページをXML形式で提供する。

(2) 各商品カタログのXML記述には「価格」が含まれている。

(3) 各商品カタログのXML記述には「発注手続き」に関するフォームページへのURLが含まれている。

(4) 各商品カタログの表示部に対してドラッグイベントハンドラを用意し、そのハンドラでは当該商品カタログのXML記述をクリップボードへコピーする処理を入れる。

【0160】図22に、モール参加規約に則ったショッピングページの一例を示す。

【0161】図22(a)はウェブページのソースの例であり、この例ではルートタグ<body>から<body/>までの記述範囲(2901)が、1つの商品カタログのソースデータに相当する。商品カタログは、ショッピングページの作者がショッピングページにおいて独立に取り出されることを意図しているショッピングページの部分的な情報資源であり、ショッピングページにおいて1つ以上存在する。

【0162】図22(a)に例示するように、商品データには、著者等の情報(<author>、<title>、<isbn>)の他に、価格に関する情報(<p

26

rice price="yes" unit="yen">2000</price>)および発注手続に関する情報(<order href=". /order.cgi?id=111"/>)が含まれている。

【0163】図22(b)は、(a)のショッピングページを表示した例であり、表示イメージ2902が商品データ2901の表示領域を表している。

【0164】図22(c)は、商品データ(2901)の表示部に対するドラッグイベントハンドラを表している。このイベントハンドラは、この表示部に対するドラッグイベントが検出されると、その商品データを出力する(当該商品データをクリップボードへコピーする)働きを行う。このイベントハンドラは、第1の実施形態のように「タグ情報」に基づいてショッピングページのデータから該当するものを抽出するものであってもよいし、あるいはこのイベントハンドラ内に出力すべきデータが作り込まれているものであってもよい。

【0165】なお、イベントハンドラが商品データを出力する代わりに、商品データのURLを出力し、クライアント装置がそのURLにページ要求することによって、商品データ取得するように構成することも可能である。

【0166】HTMLページで提供する場合は、例えば、次のようなページ構成が考えられる。

(1) ショッピングページをASPなどのサーバスクリプトページで記述する。また、そのページには各商品に関するXMLデータが含まれている。

(2) 各商品のXMLデータには「価格」が含まれている。

(3) 各商品のXMLデータには「発注手続き」に関するフォームページへのURLが含まれている。

(4) ASP内のスクリプトでは、ページ内のXMLデータをHTMLに変換する。この変換ルールは、XSLとして記述することも可能である。

(5) この際、各商品表示部に対して、当該XMLデータをクリップボードへコピーするドラッグイベントハンドラを順にセットする。XMLデータは、ハンドラプログラム内において、文字列定数として展開されて埋め込まれている。

【0167】本実施形態で例示するパーソナルショッピングモールシステムの主な機能は、以下のようである。

(1) モール参加規約に則った任意のショッピングサイトから任意の商品カタログを、システム内に用意されたショッピングカートに、ドラッグアンドドロップすることにより、その商品カタログを(ページ部品にして)取り込むことができる。

(2) システムが生成したショッピングカートページ内で、システム内に取り込まれた商品カタログ群に関して、次の処理を実行できる。

・比較

- ・合計金額見積もり
- ・個数設定
- ・一括発注

図23に、本実施形態のクライアント装置（計算機）上で実現されるパーソナルショッピングモールシステムの全体構成例を示す（なお、サーバ装置も図示している）。

【0168】顧客のウェブクライアント上に設置されたパーソナルショッピングモールシステム3001は、所望のショッピングサイトに配置されているウェブサーバ3000と、HTTPを介して接続可能であり、モール参加規約則った所望のショッピングページ3820を取得することができる。なお、第1の実施形態と同様、ページ受信モジュール280およびまたはページ発信モジュール282を備えてもよい。

【0169】パーソナルショッピングモールシステム3001は、大きく分けて、ショッピングサイトブラウザ3100と、ショッピングカートブラウザ3200から構成されている。

【0170】ショッピングサイトブラウザ3100は、外部サイトの複数のショッピングサイトを切り替え表示する機能を持っている。

【0171】ショッピングカートブラウザ3200は、ショッピングサイトブラウザ3100にて表示されたショッピングページの中からユーザが選択した商品カタログをショッピングカートページ（3824）に列挙表示するとともに、さまざまなショッピングカートの機能を提供する。

【0172】本システムの基本的な利用方法としては、任意の外部のショッピングサイトをショッピングサイトブラウザ3100上で開き、そのページ上の任意の商品をドラッグアンドドロップ操作3800を用いてショッピングカートブラウザ3200上へピックアップすることを想定している。

【0173】図24に、この商品カタログ選択時（ドラッグアンドドロップ操作時）の処理の流れの概要を示す。すなわち、ユーザから要求があった場合には、該当するサーバにアクセスしてショッピングページをダウンロードし（手順2401）、ダウンロードされたウェブページを解釈して、該ウェブページをCRT等の表示装置に表示する（手順2402）。なお、既にダウンロードされ保持されているショッピングページが要求された場合には、ダウンロードせずに、保持されているショッピングページを表示してもよいが、保持してから一定時間以上経過しているページは（更新されている可能性がある）これを再度ダウンロードするように構成すると好ましい。また、ウェブページとして記録媒体により取得したもの（例えば、電子店舗事業者より郵送されてきたもの）を併用することも可能である。さらに、第1の実施形態と同様に、ウェブページを表示する際には、

商品カタログについては、それがページ部品にできることをユーザに分かるように表示等してもよい。次に、ユーザ操作によりドラッグされた商品カタログを抽出しこれをページ部品にして（手順2403）、ユーザ操作によりドロップされたショッピングカートページに挿入し（手順2404）、ショッピングカートページを（再）表示する（手順2405）。

【0174】以下、それぞれの機能およびシステム構成について説明する。

【0175】ショッピングサイトブラウザ3100は、複数のショッピングサイトを表示することができ、またブラウザ内でのドラッグ操作によって各商品をピックアップすることができる。本例では、この商品のピックアップ機能は、各ショッピングサイトが用意するウェブページに任せているため、ショッピングサイトブラウザ3100はピックアップに関知しないで済む。

【0176】さて、対象とするショッピングサイトは、一般的なウェブブラウザのようにURLを直接入力したり、ハイパーリンクをたどることにより訪問することができるのに加え、ショッピングページデータベース3101から選択することによって訪問することもできる。また、複数のショッピングサイトを選択した場合は、それらのサイトは同時に開かれタブなどによって切り替えることができる。また、立ち寄ったショッピングサイトのURLは、自動または指定によって、ショッピングページデータベース3101に登録することができる。また、データベース内において、それらURLのグループ化やカテゴリやグルーピングなどの整理をすることにより、ショッピングページやサイト同士を関連付けることができる。データベース登録されたショッピングサイトは、次回からは選択するだけで訪問することができるのに加え、関連付けられたショッピングサイト群は、そのグループ名を選択するだけで同時に開くことができる。このショッピングサイト同士を関連付ける機能を利用することにより、ユーザが好むショッピングサイトの組み合わせのショッピングモールを仮想的に無限のパターン数で作ることができる。

【0177】ショッピングカートブラウザ3200は、システムに内蔵された、ショッピングカートXSLモジュール3210と、ショッピングカートXMLモジュール3220を含んで構成される。また、後述する読込保存モジュール3230を更に含んで構成されるようにしてもよい。

【0178】ショッピングカートXSLモジュール3210は、商品発注エンジン3211、合計金額見積もりエンジン3212、商品ソートエンジン3213、ページ部品挿入エンジン3214、商品並べ替えエンジン3215、商品個数変更エンジン3216の6つの機能モジュールによって、ショッピングカートブラウザの各機能を提供する。なお、これらのエンジンは、XSLの出

力（フォーマテイングオブジェクト）となるHTMLに付属するクライアント側プログラム（例えば、JavaScript、VBScript、Applet、ActiveXなど）として実装される。

【0179】ショッピングカートXMLモジュール3220は、現在システムが格納している各商品カタログに対応する商品データの集合およびその順序を管理しており、ショッピングカートXSLモジュール3210に含まれる各機能モジュールの共有データとなっている。

【0180】ショッピングカートブラウザ3200は、ショッピングサイトブラウザ3100からドロップされた商品データ3822をショッピングカートページ（3824）上で一覧表示するとともに、例えば、比較、合計金額見積もり、個数設定、一括発注などのさまざまなショッピングカートの機能を提供する。これらの機能は、ショッピングカートページ上に配置されたボタンなどのWidgetを操作することによって利用することができる。

【0181】具体的には、例えば、商品データ3822をショッピングカートブラウザ3200にドロップすると、ページ部品挿入エンジン3214がドロップ位置を検出し、ショッピングカートXMLモジュール3220の適切な位置に当該商品データ3822を挿入し、ショッピングカートブラウザ3200のショッピングカートページ（3824）を再表示する。このとき、合計金額見積もりエンジン3212が自動的に合計金額を計算し直す。また、ユーザ操作3801に応じて商品のソートや商品個数の変更を行うことができる。これらの機能は、商品ソートエンジン3213および商品個数変更エンジン3216によって実行される。また、購入ボタンを押すなどのユーザ操作3801に応じてそれぞれの商品に対するフォームページである、発注書3900を各ウェブサーバに対してポストする。この機能は、商品発注エンジン3211によって実行される。

【0182】また、オプションとして読込保存モジュール3230が組み込まれている場合は、上述のショッピングの作業の状態をファイルとして保存することにより、作業の中断・再開ができる。なお、詳細については、後述する。

【0183】次に、ショッピングサイトブラウザ3100についてより詳しく説明する。

【0184】ショッピングサイトブラウザ3100は、内部にショッピングページデータベース3101を持っているだけの単純な構造である。そこで、ここでは、ショッピングページデータベースの機能を具体的に説明する。実装は一般的なDBMSと同じでよい。

【0185】ショッピングページデータベース3101は、ショッピングサイトやページのURLを木構造で管理するデータベースである。

【0186】図25に、ショッピングページデータベ

ス3101に登録されたURLの一例を示す。「データベースのコンテンツ」のうち、ハッチングで示したノードは、URLが保持するノード（これをブックマークと呼ぶ）であり、木構造の葉でなければならない。その他のノードは、それらブックマークをグループ化するためのノード（これをフォルダと呼ぶ）であり、ブックマークおよびフォルダを子ノードを持つことができる。

【0187】ショッピングページデータベース3101は、ユーザのコマンドにより、URLを任意のフォルダに登録することができる。また、自動登録モードをオンにしておくと、ユーザの明示的なコマンドなしで、訪問したURL（あるいは商品の操作を行ったURL）の履歴を特定フォルダに自動的に登録することができる。また、ユーザの構成編集系コマンド群によって、フォルダを新規作成したり、各ノードを、移動、コピー、削除することができる。

【0188】また、ショッピングサイトブラウザ3100でショッピングサイトを閲覧する際は、ショッピングページデータベース3101から任意のノードを選択して開くことができる。対象とするノードがブックマークであれば、1箇所のウェブページだけが開かれるが、フォルダである場合、子孫のすべてのブックマークのウェブページが同時にブラウザ上に開かれ、それらはタブなどによって任意の順序で切り替えることができる。

【0189】次に、ショッピングカートブラウザ3200についてより詳しく説明する。

【0190】ショッピングカートブラウザ3200は、システムに内蔵された、ショッピングカートXSLモジュール3210と、ショッピングカートXMLモジュール3220から構成される。

【0191】ショッピングカートブラウザ3200が提供する機能モジュールは、ショッピングカートXSLモジュール3210内にあり、ショッピングカートXMLモジュール3220は、それら機能モジュールの共有データとしての位置付けを果たしている。そこで、ここでは、ショッピングカートXSLモジュール3210に含まれる、商品発注エンジン3211、合計金額見積もりエンジン3212、商品ソートエンジン3213、ページ部品挿入エンジン3214、商品並べ替えエンジン3215、商品個数変更エンジン3216の6つの機能モジュール別に、それらの機能とアルゴリズムについて説明する。

【0192】まず、ページ部品挿入エンジン3214について説明する。

【0193】図26に、ページ部品挿入エンジン3214の構成例を示す。図26に示されるように、ページ部品挿入エンジン3214は、ドロップイベント監視プログラム32141、ルートノード特定プログラム32142、商品カタログ挿入プログラム32143、ウェブページ再表示プログラム32144から構成されてい

31

る。これらの機能により、ショッピングサイトブラウザ3100からのドロップによって、ユーザが希望する位置へ当該商品カタログが挿入される。

【0194】図27に、この場合の処理の一例を示す。以下、このアルゴリズム例を順に説明する。

【0195】手順2701において、ドロップイベント監視プログラム32141は、XSL処理後のショッピングカートHTML(3824)内において、それぞれの商品カタログおよびbodyタグに対してドロップイベントハンドラが埋め込まれている。もし新規の商品カタログがドロップされた場合、そのドロップされた場所に存在する商品カタログのドロップイベントハンドラが起動されるか、そのドロップされた場所に商品カタログが存在しない場合は、bodyタグに埋め込まれたドロップイベントハンドラが起動される。前者のドロップイベントハンドラを「位置指定型ドロップイベントハンドラ」、後者のドロップイベントハンドラを「デフォルトドロップイベントハンドラ」と呼ぶものとする。位置指定型ドロップイベントハンドラが起動された場合は、手順2702へ進み、デフォルトドロップイベントハンドラが起動された場合は、手順2704へ進む。

【0196】手順2702において、位置指定型ドロップイベントハンドラは、イベント発生源のHTMLからそれに該当するXMLの文書領域のルートノードをショッピングカートXMLモジュール3220から特定する。特定の方法の一つとしては、表示に際してXMLからHTMLへ変換するときに、各商品カタログのXML文書のルートノードおよび変換後のHTMLのルートノードにユニークな値のID属性を付けておき、特定するときはそのID属性値を元にサーチするという方法がある。

【0197】手順2703において、特定されたイベント発生源のXMLルートノードの1つ上の兄となる位置に、ドロップされた商品カタログのXML文書を挿入する。

【0198】手順2704において、ショッピングカートXMLモジュール3220のうち、商品カタログ群を挿入するノード位置はあらかじめシステム内で決められている。デフォルトドロップイベントハンドラは、その位置に対して、ドロップされた商品カタログのXML文書を挿入する。

【0199】手順2705において、ショッピングカートXMLモジュール3220が更新されたので、それに対してショッピングカートXSLモジュール3210を適用して、ウェブページを再表示する。

【0200】次に、商品並べ替えエンジン3215について説明する。

【0201】図28に、商品並べ替えエンジン3215の構成例を示す。図28に示されるように、商品並べ替えエンジン3215は、大きく、ドラッグ処理プログラ

32

ム3218、ドロップ処理プログラム3219から構成される。また、ドラッグ処理プログラム3218は、ドラッグイベント監視プログラム32181、ドラッグ実行処理プログラム32182から構成されており、ドロップ処理プログラム3219は、ドロップイベント監視プログラム32191、ルートノード特定プログラム32192、商品カタログ挿入プログラム32193、ウェブページ再表示プログラム32194から構成されている。これらの機能により、ショッピングカートブラウザ3200内でのドラッグアンドドロップによって、商品カタログの順序を並べ替えることができる。

【0202】まず、図29に、ドラッグ時の処理の一例を示す。

【0203】手順3901において、ドラッグイベント監視プログラム32181は、XSL処理後のショッピングカートHTML(3824)内において、商品カタログごとにドロップイベントハンドラとして埋め込まれている。もしショッピングカート内の商品カタログのいずれかがドラッグされた場合、その商品カタログに割り当てられていたドロップイベントハンドラが起動される。

【0204】手順3902において、ドロップイベントハンドラでは、ドラッグ処理を実行するが、これがドラッグ処理実行プログラム32182にあたる。ドラッグ処理実行プログラム32182は、イベント発生源のHTMLからそれに該当するXMLの文書領域のルートノードをショッピングカートXMLモジュール3220から特定する。特定の方法の一つとしては、表示に際してXMLからHTMLへ変換するときに、各商品カタログのXML文書のルートノードおよび変換後のHTMLのルートノードにユニークな値のID属性を付けておき、特定するときはそのID属性値を元にサーチするという方法がある。また、この特定されたノードに対して特定の属性(例えば、dragging="yes"など)を付けておき(この属性をドラッグ属性と呼ぶものとする)、プログラム内部の特定のグローバルなフラグ変数(例えば、bDragging)をTrueにセットする(この変数を並べ替えモードフラグと呼ぶものとする)。これらの情報は、ドロップ処理プログラム3219で利用される。

【0205】次に、図30に、ドロップ時の処理の一例を示す。この処理例は、ページ部品挿入エンジン3214による処理例(図27)とほぼ同じであり、異なるのは手順3913および手順3914での挿入時に、挿入元となる商品カタログを消去する作業が加わっているだけである。実装においてもページ部品挿入エンジン3214の機構を流用することができる。

【0206】手順3911において、ドロップイベント監視プログラム32191は、XSL処理後のショッピングカートHTML(3824)内において、それぞれ

10

20

30

40

50

の商品カタログおよびbodyタグに対してドロップイベントハンドラが埋め込まれている。もし新規の商品カタログがドロップされた場合、そのドロップされた場所に存在する商品カタログのドロップイベントハンドラが起動されるか、そのドロップされた場所に商品カタログが存在しない場合は、bodyタグに埋め込まれたドロップイベントハンドラが起動される。前者のドロップイベントハンドラを位置指定型ドロップイベントハンドラ、後者のドロップイベントハンドラをデフォルトドロップイベントハンドラと呼ぶものとする。位置指定型ドロップイベントハンドラが起動された場合は、手順3912へ進み、デフォルトドロップイベントハンドラが起動された場合は、手順3914へ進む。

【0207】手順3912において、位置指定型ドロップイベントハンドラは、イベント発生源のHTMLからそれに該当するXMLの文書領域のルートノードをショッピングカートXMLモジュール3220から特定する。特定の方法の一つとしては、表示に際してXMLからHTMLへ変換するときに、各商品カタログのXML文書のルートノードおよび変換後のHTMLのルートノードにユニークな値のID属性を付けておき、特定するときはそのID属性値を元にサーチするという方法がある。

【0208】手順3913において、特定されたイベント発生源のXMLルートノードの1つ上の兄となる位置に、ドロップされた商品カタログのXML文書を挿入する。このとき、ドラッグ処理プログラム3218によってセットされる並べ替えモードフラグがTrueならば、ドラッグ属性の付いているノードを探し、そのノード以下を消去する。

【0209】手順3914において、ショッピングカートXMLモジュール3220のうち、商品カタログ群を挿入するノード位置はあらかじめシステム内で決められている。デフォルトドロップイベントハンドラは、その位置に対して、ドロップされた商品カタログのXML文書を挿入する。このとき、ドラッグ処理プログラム3218によってセットされる並べ替えモードフラグがTrueならば、ドラッグ属性の付いているノードを探し、そのノード以下を消去する。

【0210】手順3915において、ショッピングカートXMLモジュール3220が更新されたので、それに対してショッピングカートXMLモジュール3210を適用して、ウェブページを再表示する。

【0211】次に、商品個数変更エンジン3216について説明する。

【0212】商品個数変更エンジン3216は、ショッピングカート上の各商品に関して発注する個数を変更する機能を提供する。これを実現するためには、モール参加規約に、次のルールを追加する。・発注手続きのURLに関して、個数を指定する特定のパラメータ記述をサ

ポートすること。

【0213】具体的には、ある商品に関する発注URLが“http://www.hoge.com/order.cgi?id=111”であるとして、2個発注する場合は、“http://www.hoge.com/order.cgi?id=111&num=2”とするなどのルールである。ここでは、モール参加規約は、“num=”で商品発注個数を記述するというルールを追加採用するものとする。

10 【0214】商品個数変更エンジン3216は、JavaScriptやVBScriptなどのDOMをアクセスできるクライアントサイドプログラムで実装するのが最も簡単である。ここでは、そのようなプログラムシステムで実装する。

【0215】商品個数変更エンジン3216は、2つの処理に分かれる。

20 【0216】一つは、ショッピングサイトブラウザ3100からショッピングカートブラウザ3200にドロップされたときの処理である。ドロップするときにはまだ発注個数は決定していないので、商品記述のXMLには個数に関する記述が含まれていない。したがって、商品カタログのXMLの中に、<number>1</number>という文書を挿入する。

30 【0217】次は、個数変更の処理である。個数変更は、ウェブページ上のテキストボックスに対するユーザー書き込みによって行われる。テキストボックスの値が変化すると、個数変更の処理プログラムが起動される。この処理プログラムは、テキストボックスの値を読み取り、ショッピングカートXMLモジュール3220内の該当する個数記述(number要素)を特定して要素値を変更する。ただし、該当するXMLの個数記述を特定する方法は、ページ部品挿入エンジン3214と同じ方法(IDによってサーチする方法)を使う。

【0218】次に、合計金額見積もりエンジン3212について説明する。

40 【0219】合計金額見積もりエンジン3212は、JavaScriptやVBScriptなどのDOMをアクセスできるクライアントサイドプログラムで実装するのが最も簡単である。ここでは、そのようなプログラムシステムで実装する。

【0220】ショッピングカートXMLモジュール3220は、各商品において価格の情報および発注個数を持っている。合計金額見積もりエンジン3212は、出力するHTMLのうち合計金額を表示するテキストボックスなどのタグを処理するときに、ショッピングカートXMLモジュール3220の中の各商品の価格と発注個数を読み取って合計を計算し結果をHTMLの当該タグの値として書き込むように作られている。したがって、ウェブページを(再)表示する際に実行される。

50 【0221】ただし、各商品の価格をXMLから取り出

すためには要素名や属性名など何らかの手がかりが必要であるため、元のショッピングサイト側は、価格のXML記述に関するルールにしたがって商品ページを提供していなければならない。

【0222】次に、商品ソートエンジン3213について説明する。

【0223】商品ソートエンジン3213は、JavaScriptやVBScriptなどのDOMをアクセスできるクライアントサイドプログラムで実装するのが最も簡単である。ここでは、そのようなプログラムシステムで実装する。

【0224】ショッピングカートXMLモジュール3220は、各商品において、さまざまなパラメータを持っている。商品ソートエンジン3213は、各商品に含まれる項目をすべて取り出して、任意の項目でソートできるようになっている。商品ソートエンジン3213には、ユーザ操作のためのWidgetとして、ソート項目を選択するセレクトとソートを実行するボタンが用意されており、処理は大きく二つに分かれる。

【0225】まず、処理の一つは、セレクトで選択できる項目を作成である。これは、出力するHTMLのうち当該セレクトのタグを処理するとき、すなわちウェブページの（再）表示時に実行され、各商品が持つすべての要素名をリストアップしてその結果をセレクトに登録する。

【0226】もう一つの処理は、ソート自身である。XSL言語は、ソート命令を持っているので、商品ソートエンジン3213は、その機能を利用する。具体的には、ショッピングカートXSLモジュール3210のXSLソース内の商品カタログを列挙する命令タグに対して、セレクトで現在選択されている項目でソートするようなソート命令を上書きする。すなわち、自己書き換えを行う。そして、改めてウェブページを再表示する。これにより、ソートされたウェブページが表示される。

【0227】また、商品表示の順序をドラッグアンドドロップで並び替えることができるようにしてもよい。この場合には、まず、ユーザにより商品カタログがドラッグされると自分自身の内容をクリップボードにコピーする、そして、その商品カタログがドロップされるとその位置にその商品カタログをクリップボードからコピーするとともに、元の商品カタログを消す。

【0228】次に、商品発注エンジン3211について説明する。

【0229】この場合には、ショッピングサイトが提供する商品カタログ内に購入手続きへの実行エン트리ポイントが埋め込まれているものとする。

【0230】商品発注エンジン3211は、JavaScriptやVBScriptなどのDOMをアクセスできるクライアントサイドプログラムで実装するのが最も簡単である。ここでは、そのようなプログラムシステム

ムで実装する。

【0231】商品発注エンジン3211は、ショッピングカートブラウザ3200に含まれる各商品を一括購入する機能を提供する。具体的には、ショッピングカートブラウザ3200で表示されているウェブページの一括購入に関するボタンを押すと、各商品に関して発注の手続きが含まれるフォームページを表示しているダイアログボックスがポップアップされる。それらすべてのダイアログボックス上でフォームページを送信すると、各商品がショッピングカート上で指定した個数で発注される。

【0232】処理手順は、次のようになる。まず、一括購入ボタンのクリックイベントを監視し、押された時に、各商品について、その発注手続き用のURLとその発注個数を取り出す。続いて、各商品について、取り出されたURLの末尾に対して、“&num=2”など（2個の場合）の個数情報を付け加えて、購入手続きをキックする。そして、取得した当該URLのウェブページ（フォームページ）を表示するダイアログボックスを開く。後は、それぞれのフォームページにおいて、ユーザが必要事項を入力して手続を行う。

【0233】次に、読込保存モジュール3230について説明する。

【0234】読込保存モジュール3230は、パーソナルショッピングモールで行った作業状況をファイルへ保存したり、保存されたファイルを読み込んで前回の作業状況を復活させ、作業の続きを行うことを可能にする。

【0235】保存の方法としては、例えば、ショッピングカート単位で保存する方法と、部品単位で保存する方法の2種類の保存方式があり、それぞれ効果と実装が異なる。

【0236】まず、ショッピングカート単位で保存・読込する方法について説明する。この場合、保存する際には、ショッピングカートXMLモジュール3220の内容、およびショッピングサイトブラウザ3100で開いているURLのリストを書き出す。逆に、リジュームさせる場合は、それらを読み込んで状態へ反映させる。これにより、作業の続きを行うことが可能となる。

【0237】次に、商品単位で保存・読込する方法について説明する。この場合、保存する際には、ショッピングカートXMLモジュール3220の内容を各商品単位に分解し、指定された商品についてもしくはすべての各商品について商品情報を書き込む。ただし、商品情報には、その商品カタログがもともと掲載されていたウェブページのURL情報が含まれていなければならない。逆に、リジュームさせる場合は、まずユーザが対象とする商品情報を複数選択する。選択された商品情報にそれぞれを、ショッピングカートXMLモジュール3220に対して追加する。それと同時に、その商品情報に含まれるURLをショッピングサイトブラウザ3100に追加

37

する。これにより、指定された商品情報のみがショッピングカートブラウザ3200で登録され、またそれらの商品に関するウェブページのみがショッピングサイトブラウザ3100で開かれる結果となる。

【0238】なお、保存の方法については、例えば、複数種類備えておき、ユーザが適宜選択して使用可能としてもよいし、いずれか1種類のみを備えるようにしてもよい。

【0239】次に、本パーソナルショッピングモールシステムのGUIについて説明する。

【0240】図31に、本パーソナルショッピングモールシステムの実行画面の一例を示す。

【0241】大きく二つのウィンドウがあるが、左側のウィンドウ4100がショッピングサイトのウェブページを表示しているウェブブラウザで、下部には現在同時に開いているショップを切り替えるためのタブ(4103~4106)が付いている。また、右側のウィンドウ4200が集めてきた商品の一覧(4211~4214)を表示しているウェブブラウザである。

【0242】訪れるショッピングサイトは、一般のウェブブラウザと同様URLを入力(4001)することによって選択することもできるし、ショッピングページデータベース3101からお気に入りショップとして選択(4002)することもできる。

【0243】ユーザは、複数のショッピングサイトを回って、気に入った商品があればその商品カタログ(4101)を右側のウェブブラウザでドロップする。これにより、ショッピングカートに登録される。

【0244】この作業を経て集めてきた商品群に対し、ソート条件の指定(4202)、ソート(4201)、合計値段見積もり(4003)、個数設定(図示せず)、一括発注(4204)を行うことができる。これらは、ショッピングカートページの中に配置されているwidgetを操作することで実行できる。

【0245】個数の表示・設定は、例えば、ショッピングカートブラウザのウィンドウ4200のショッピングカートページに表示されている商品カタログの領域内で行われる。

【0246】なお、ショッピングサイトブラウザのウィンドウ4100の個別のサイトのページで、個別の商品の発注ボタン4120を押して、個別に発注することももちろん可能である。

【0247】以下では、本実施形態の利点について説明する。

【0248】ショッピングサイトの連携を図る方法の一つとして、Harringtonの米国特許(USP: 5,895,454)のように構成対象のショッピングシステムのデータベース管理システムを接続する手法がある。説明の便宜上、この方法によって実現された統合システムを「サーバショッピングモールシステム」と呼

38

ぶものとし、サーバショッピングモールシステムと比較したパーソナルショッピングモールシステムの優位性について以下に説明する。

【0249】パーソナルショッピングモールシステムは、サーバショッピングモールに対して、各ショッピングサイトの構築コストに関して次の優位性がある。サーバショッピングモールシステムの場合、出店するショッピングサイトは、モールシステム用に特別にシステムを作りこむ必要がある。また、参加先のモールシステムを変更する際には、ショッピングシステムを構築しなおさなければならない。それに対し、パーソナルショッピングモールシステムの場合、対象となるショッピングサイトは、提供する商品カタログ記述に関してモール参加規約を遵守するだけでよい(特別なシステムは不要)。

【0250】サーバショッピングモールシステムにおけるショッピングサイトの出店構成は、概して固定的である。なぜなら出店構成を変更するためには、システムを再構築しなければならないからである。その出店構成が顧客の望むものではなかったとしても、顧客は出店構成を変更することはできない。それに対しパーソナルショッピングモールシステムの場合、あらかじめ与えられた出店構成というものはなく、顧客は任意のサイトの商品を組み合わせることができる。つまり、顧客は好きなショッピングサイトの商品を組み合わせるショッピングサイト先を自由に決められるので、顧客の嗜好性に合わせたショッピングサイトの組み合わせ、すなわちマイ商店街を作ることが可能である。このことは、例えば、契約などによる取引先に関する制限を持つ企業顧客、ロジスティクスや政治的問題などに関して地域的な制約を持った顧客、レアなジャンルの商品を捜し求めているなど対象商品に制約を持つ顧客などに利益をもたらすことも期待される。また、中古ショップや「売ります買います」サイトなど、草の根的なショッピングサイトを束ねるのにも便利である。また、サーバショッピングモールシステムとして構築されたポータルサイトをまたいで、商品をピックアップすることもできるので、サーバショッピングモールシステムを顧客側で補完する使い方もできる。

【0251】なお、第2の実施形態の構成例では、フォームページを取り寄せる際に該当するURLの末尾に対して“&num=2”などの個数情報を付け加えて購入手続きをキックしたが、このときには個数情報を付加せず、個数情報はユーザがフォームページで入力するようにしてもよい。また、一括購入数量によって単価が変わる商品を扱う場合、例えば、商品カタログの中に、一括購入数量と単価の関係または一括購入数量と総額の関係記述することを規約に含めるようにしてもよいし、あるいは一括購入数量毎に商品カタログを用意するようにしてもよい。また、第1の実施形態のように、スタイルを適宜設定あるいは選択可能としてもよい。また、商品

39

売買を例にとったが、もちろんホテルの部屋あるいは電車の座席の予約等、種々の契約についても適用可能である。

【0252】さらに、第1の実施形態において第2の実施形態におけるページ部品抽出方法を用いる異も可能であり、第2の実施形態において第1の実施形態におけるページ部品抽出方法を用いる異も可能である。

【0253】本発明は、上述した実施の形態に限定されるものではなく、その技術的範囲において種々変形して実施することができる。

【0254】

【発明の効果】本発明によれば、エンドユーザは、汎用的な手続きによって、任意のウェブサイトの独立可能部分同士を組み合わせ、新しいページを作成することができる。また、本発明によれば、ウェブページの提供側に対して強い制限を与えないで済む。

【図面の簡単な説明】

【図1】本発明を適用するネットワーク環境の一例を示す図

【図2】本発明の一実施形態に係るシステムの全体構成例を示す図

【図3】同実施形態のシステムの処理の流れの一例を示すフローチャート

【図4】ウェブページの一例を示す図

【図5】ページ部品の抽出手順の一例を示すフローチャート

【図6】ページ部品の内容部の作成手順の一例を示すフローチャート

【図7】ページ部品の管理情報部の作成手順の一例を示すフローチャート

【図8】ページ部品の管理情報部の作成手順の一例を示すフローチャート

【図9】ページ部品のデータ構造の一例を示す図

【図10】ページ部品の整理手順の一例を示すフローチャート

【図11】一般アプリケーションへの接続手順の一例を示すフローチャート

【図12】ページ生成手順の一例を示すフローチャート

【図13】実行画面の一例を示す図

【図14】部品単一閲覧画面の一例を示す図

【図15】部品項目表示画面の一例を示す図

【図16】部品項目表示画面の他の例を示す図

【図17】部品項目表示画面のさらに他の例を示す図

【図18】検索用入力画面の一例を示す図

【図19】スタイル選択画面の一例を示す図

【図20】スタイル選択画面の他の例を示す図

【図21】複数のページ部品を含むウェブページの表示例を示す図

【図22】モール参加規約に則ったウェブページの一例について説明するための図

40

【図23】本発明の他の実施形態に係るパーソナルショッピングモールシステムの全体構成例を示す図

【図24】同実施形態のシステムの処理の流れの一例を示すフローチャート

【図25】ショッピングページデータベースの登録内容の一例を示す図

【図26】ページ部品挿入エンジンの構成例を示す図

【図27】ページ部品挿入エンジンの処理手順の一例を示すフローチャート

10 【図28】商品並べ替えエンジンの構成例を示す図

【図29】ドラッグ処理の手順の一例を示すフローチャート

【図30】ドロップ処理の手順の一例を示すフローチャート

【図31】パーソナルショッピングモールシステムの表示画面の一例を示す図

【符号の説明】

101…クライアント装置

102…ウェブクライアントサイト

20 103…コンピュータネットワーク基盤

104～106…ウェブサーバサイト

107～109…サーバ装置

201…ウェブページ

205…表示装置

206…アプリケーション

210…第1のソフトウェアプログラム

220…第1のソフトウェアプログラム

211…ページ表示モジュール

212…ページ部品抽出モジュール

30 213…ページ部品送信モジュール

221…ページ部品受信モジュール

222…ページ部品データベース

223…ページ部品整理モジュール

224…一般アプリ接続モジュール

225…ページ生成モジュール

280…ページ受信モジュール

282…ページ発信モジュール

610…ページ部品内容抽出モジュール

710…ページ部品管理情報生成モジュール

40 810…ページ部品アプリデータ属性情報生成モジュール

1010…部品項目表示モジュール

1020…部品単一閲覧モジュール

3001…パーソナルショッピングモールシステム

3000…ウェブサーバ

3100…ショッピングサイトブラウザ

3101…ショッピングページデータベース

3200…ショッピングカートブラウザ

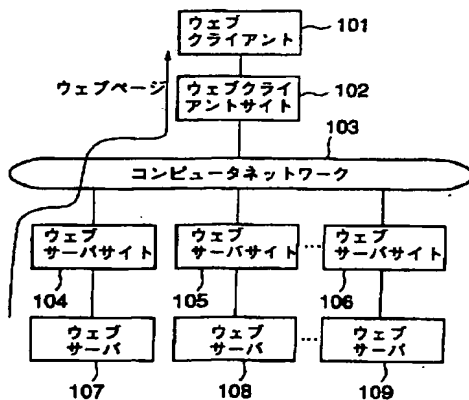
3210…ショッピングカートXSLモジュール

50 3211…商品発注エンジン

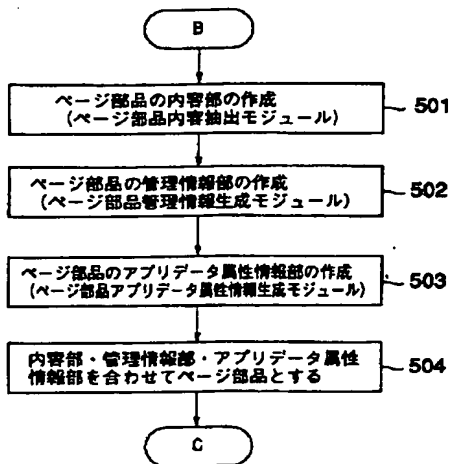
41

- 3 2 1 2…合計金額見積もりエンジン
- 3 2 1 3…商品ソートエンジン
- 3 2 1 4…ページ部品挿入エンジン
- 3 2 1 5…商品並べ替えエンジン
- 3 2 1 6…商品個数変更エンジン
- 3 2 1 8…ドラッグ処理プログラム
- 3 2 1 9…ドロップ処理プログラム
- 3 2 2 0…ショッピングカートXMLモジュール
- 3 2 3 0…読込保存モジュール

【図1】



【図5】

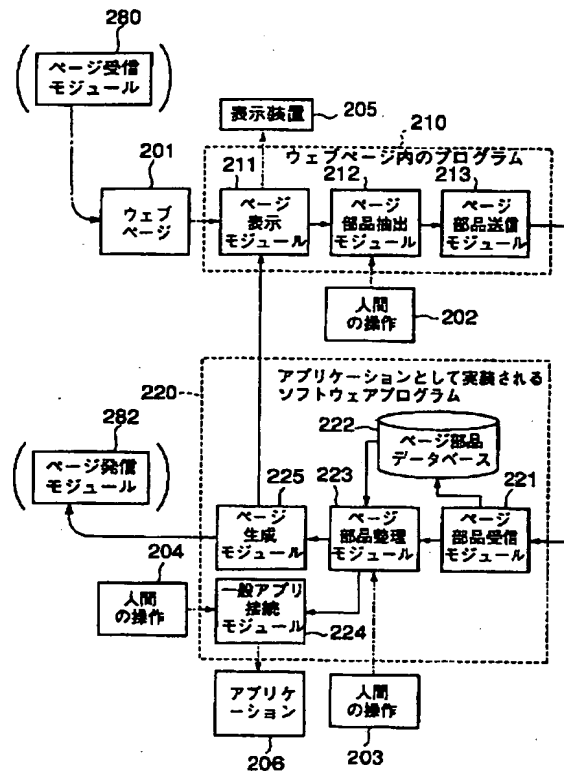


42

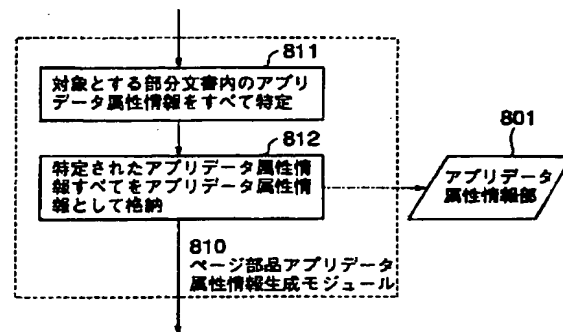
- * 3 2 1 4 1, 3 2 1 9 1…ドロップイベント監視プログラム
- 3 2 1 4 2, 3 2 1 9 2…ルートノード特定プログラム
- 3 2 1 4 3, 3 2 1 9 3…商品カタログ挿入プログラム
- 3 2 1 4 4, 3 2 1 9 4…ウェブページ再表示プログラム
- 3 2 1 8 1…ドラッグイベント監視プログラム
- 3 2 1 8 2…ドラッグ実行処理プログラム

*

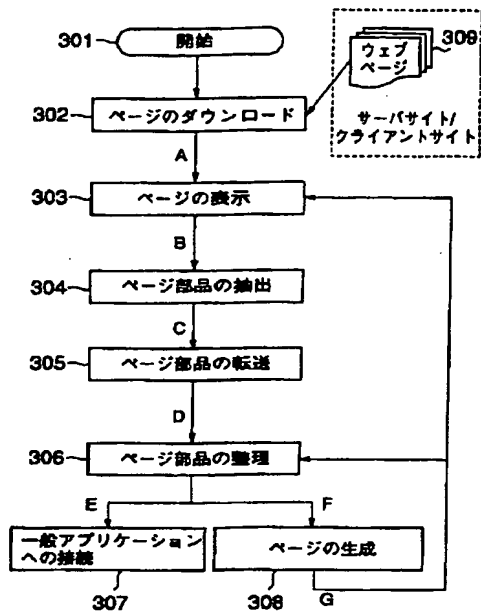
【図2】



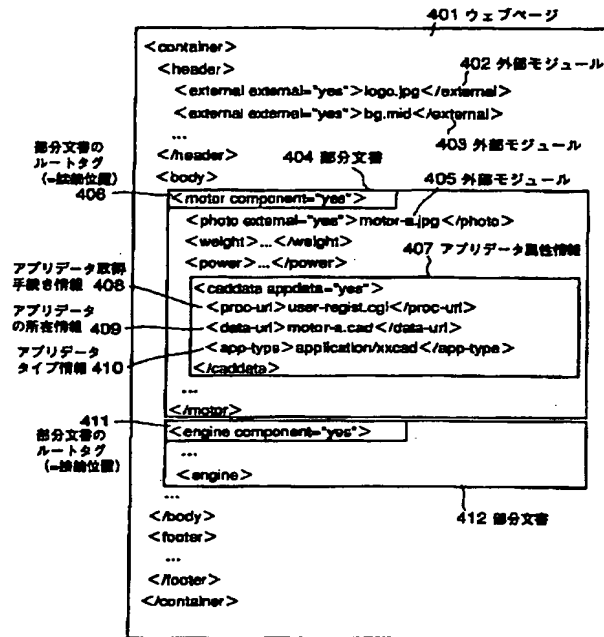
【図8】



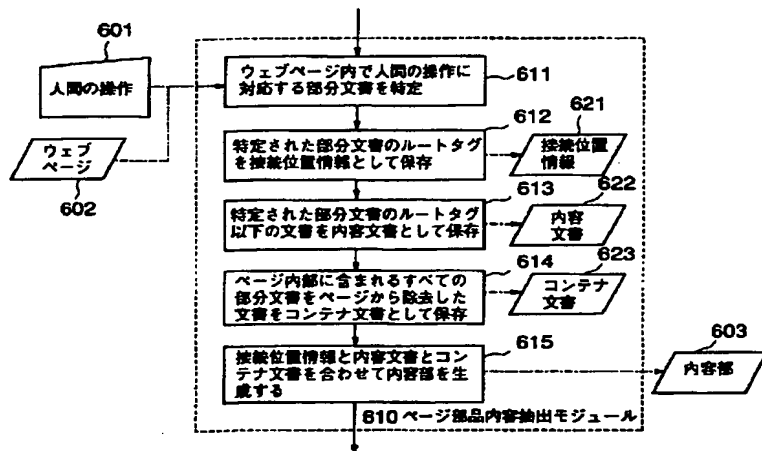
【図3】



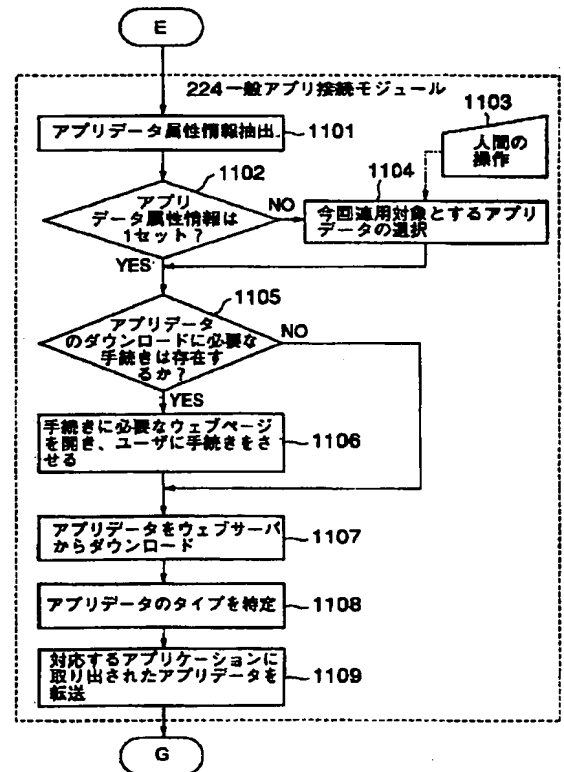
【図4】



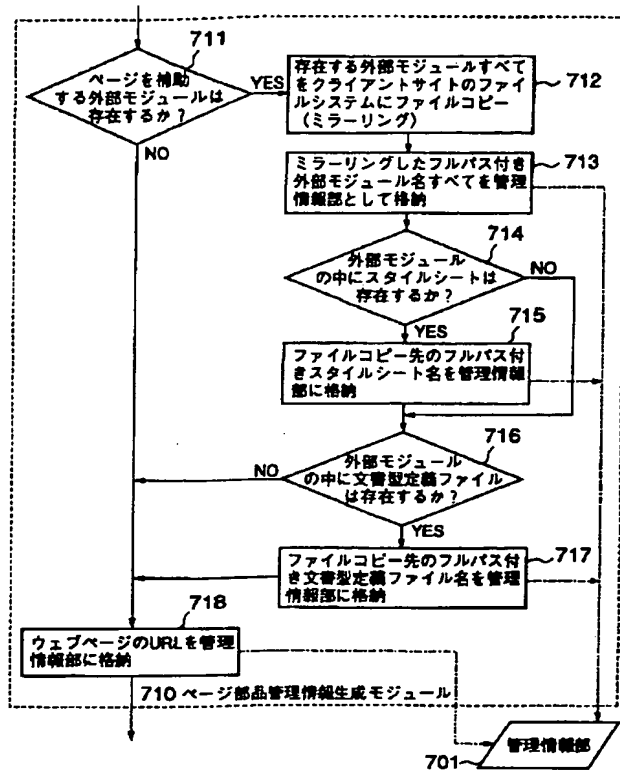
【図6】



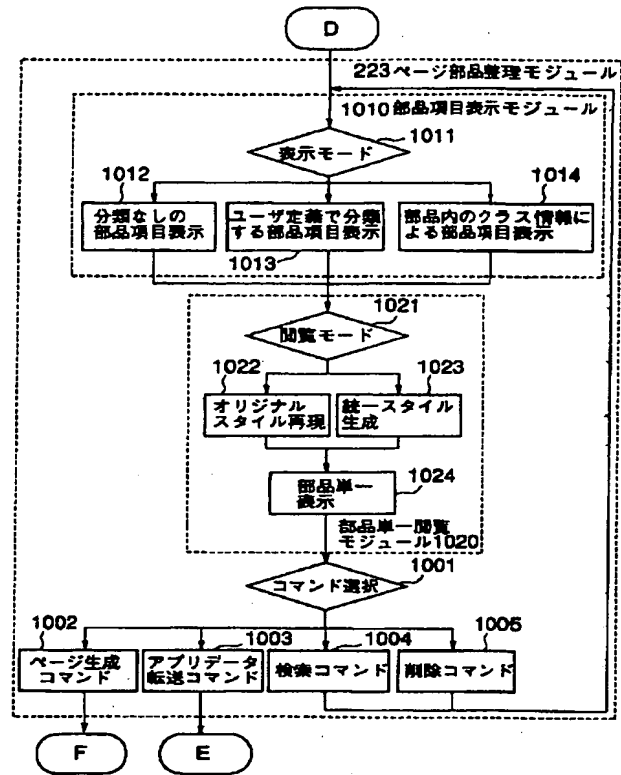
【図11】



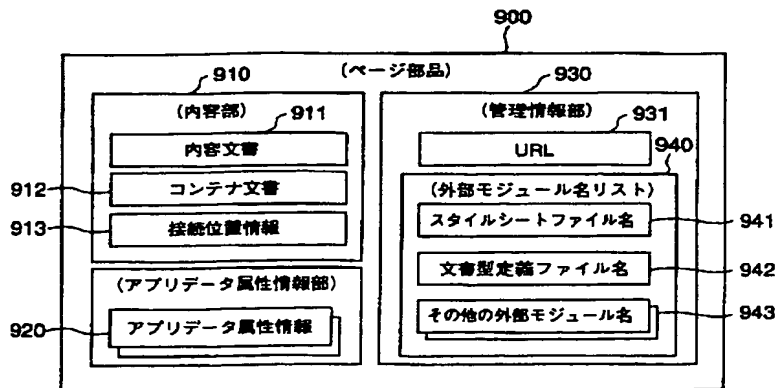
【図7】



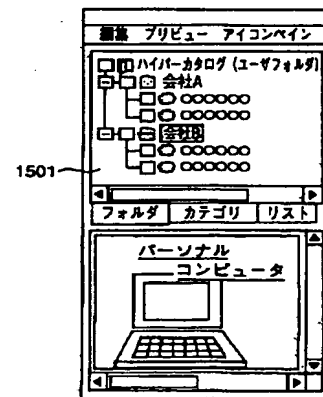
【図10】



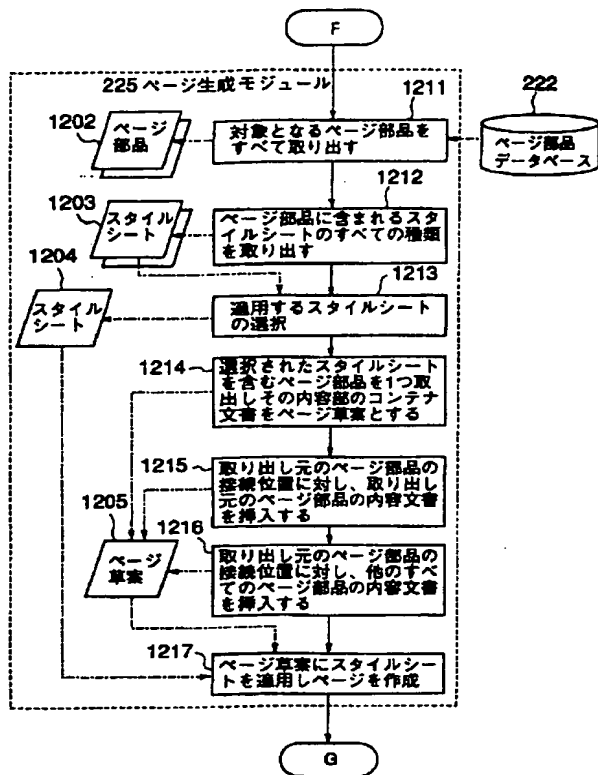
【図9】



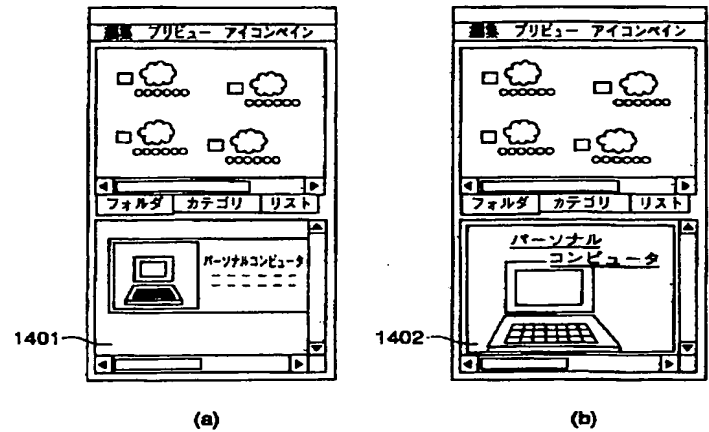
【図15】



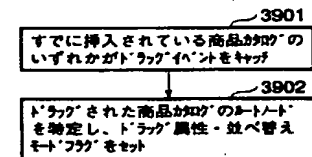
【図12】



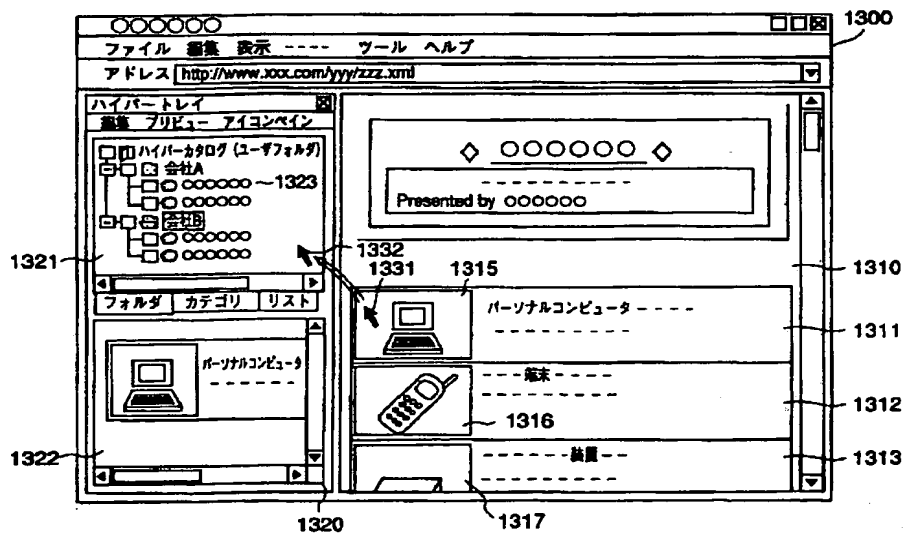
【図14】



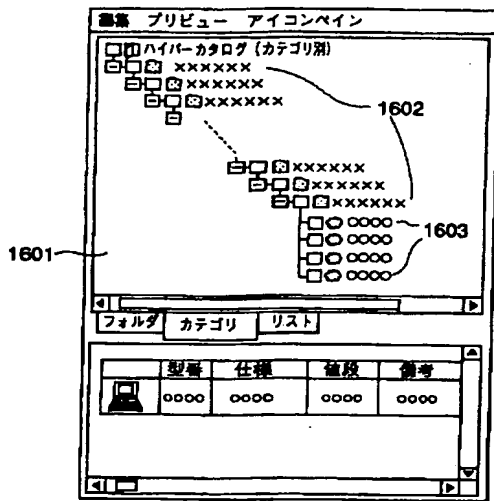
【図29】



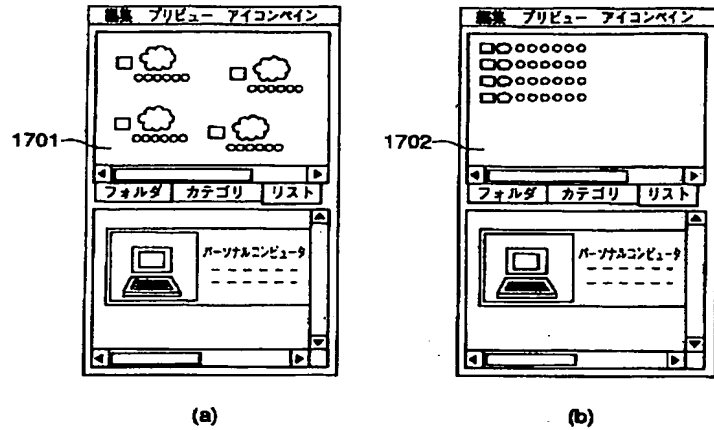
【図13】



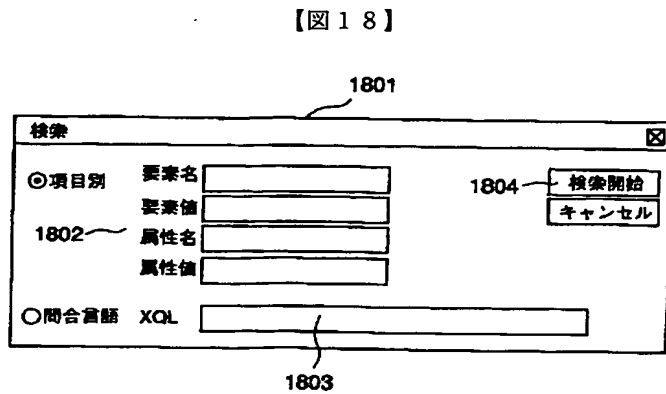
【図16】



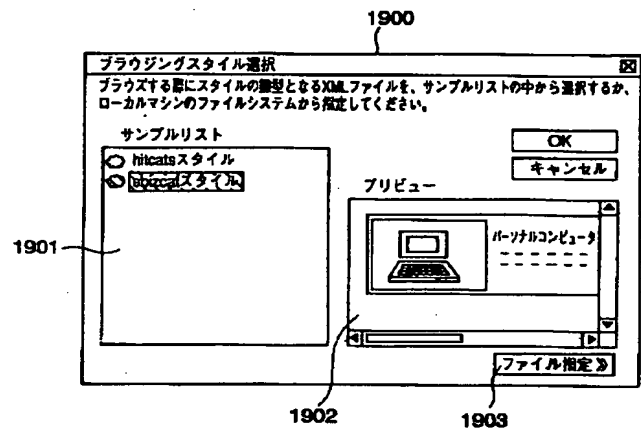
【図17】



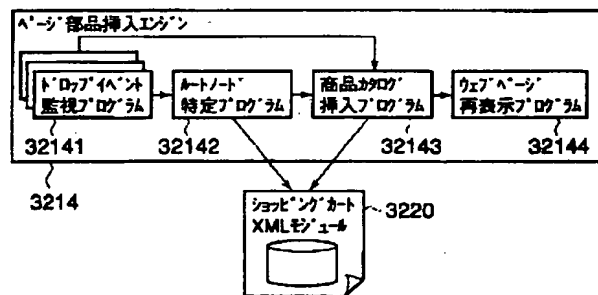
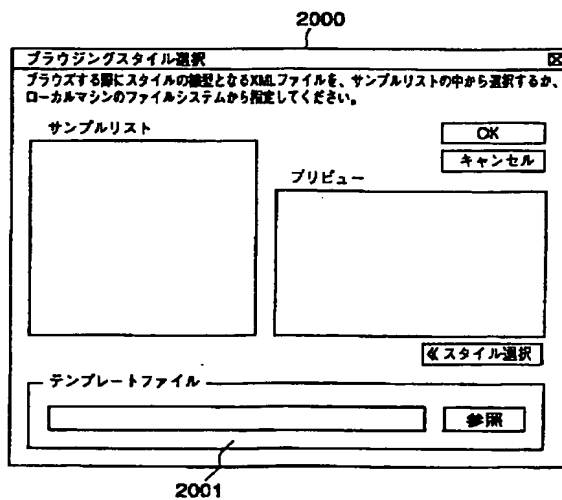
【図19】



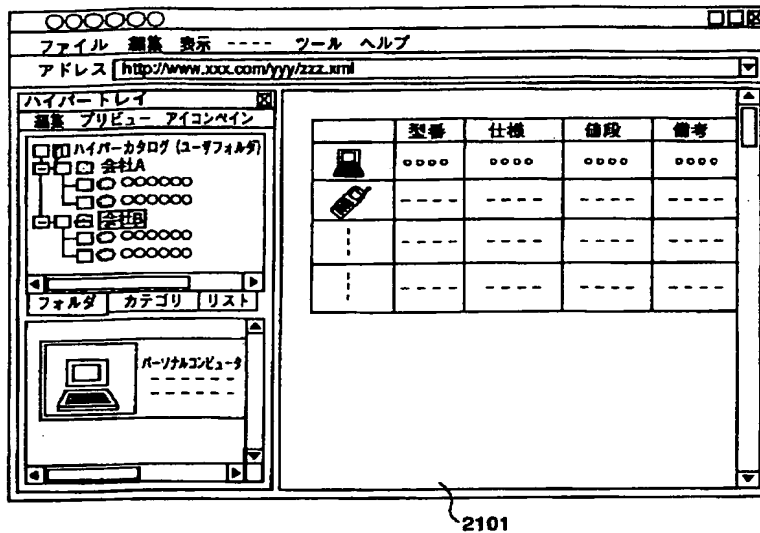
【図20】



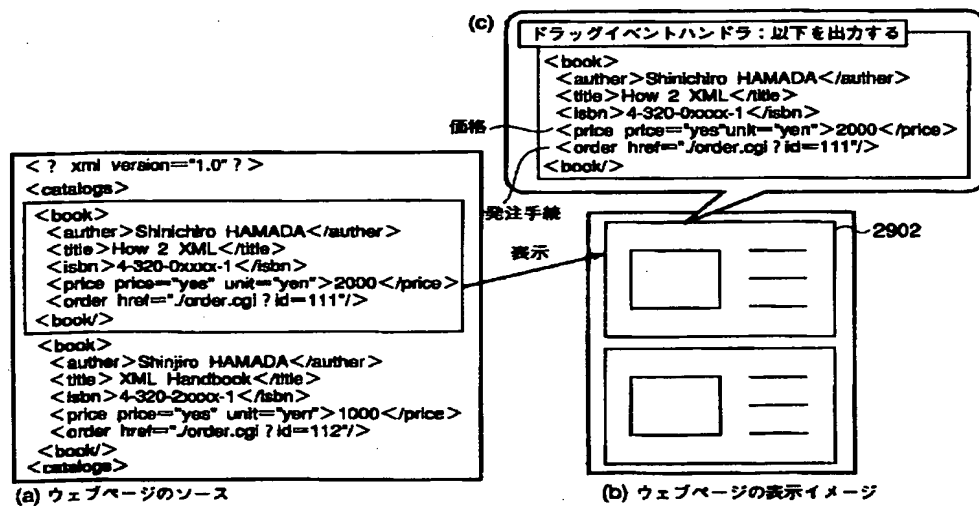
【図26】



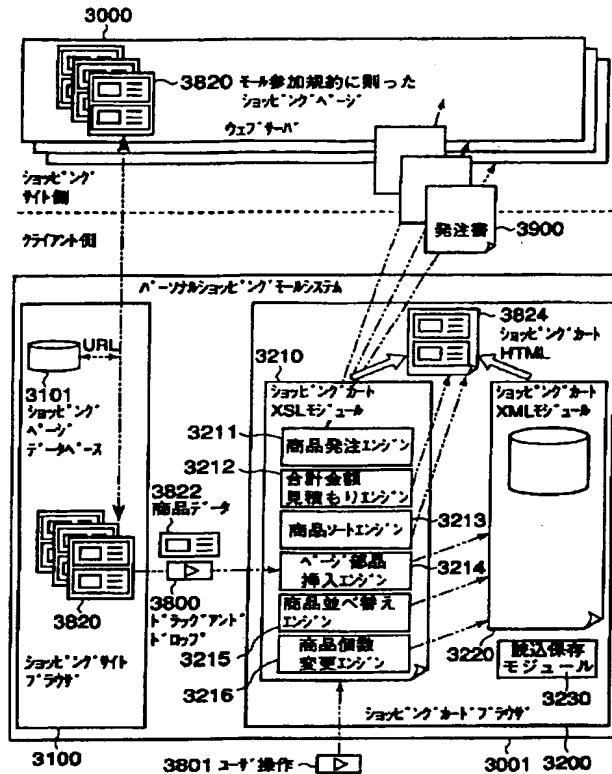
【図21】



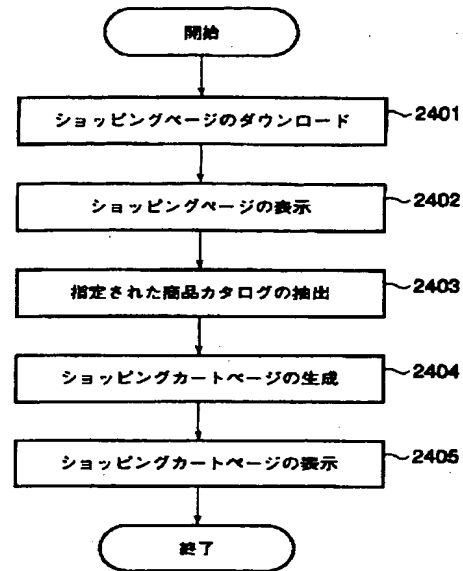
【図22】



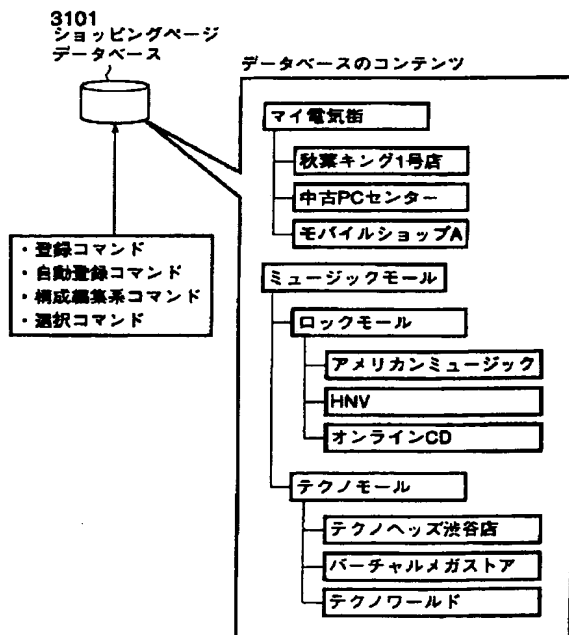
【図23】



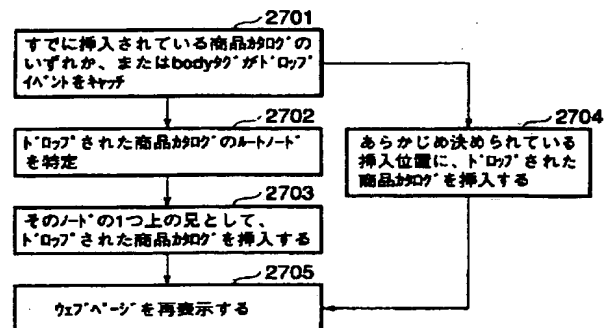
【図24】



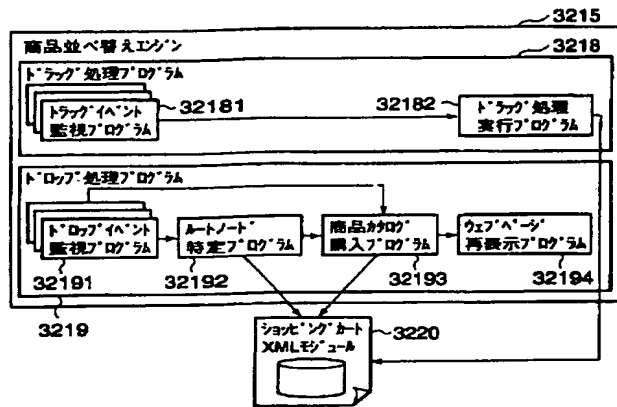
【図25】



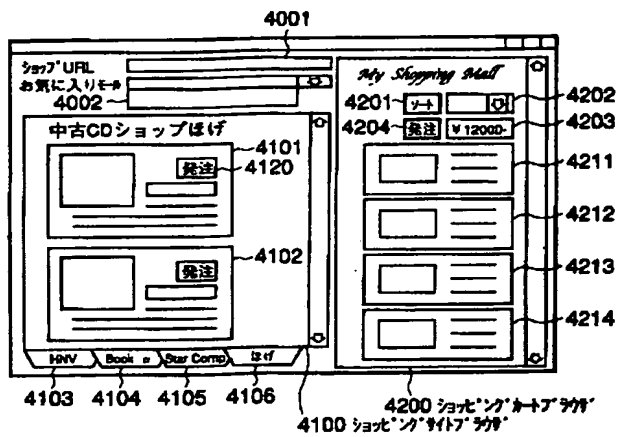
【図27】



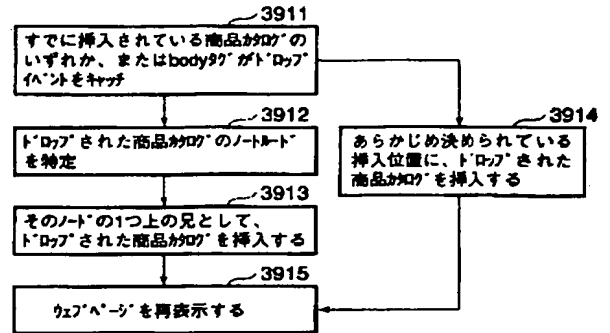
【図28】



【図31】



【図30】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ BLACK BORDERS

☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES

☐ FADED TEXT OR DRAWING

☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING

☐ SKEWED/SLANTED IMAGES

☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS

☐ GRAY SCALE DOCUMENTS

☐ LINES OR MARKS ON ORIGINAL DOCUMENT

☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY

☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.